

Colin Fyfe
Dongsup Kim
Soo-Young Lee
Hujun Yin (Eds.)

LNCS 5326

Intelligent Data Engineering and Automated Learning – IDEAL 2008

9th International Conference
Daejeon, South Korea, November 2008
Proceedings

 Springer

Commenced Publication in 1973

Founding and Former Series Editors:

Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Editorial Board

David Hutchison

Lancaster University, UK

Takeo Kanade

Carnegie Mellon University, Pittsburgh, PA, USA

Josef Kittler

University of Surrey, Guildford, UK

Jon M. Kleinberg

Cornell University, Ithaca, NY, USA

Alfred Kobsa

University of California, Irvine, CA, USA

Friedemann Mattern

ETH Zurich, Switzerland

John C. Mitchell

Stanford University, CA, USA

Moni Naor

Weizmann Institute of Science, Rehovot, Israel

Oscar Nierstrasz

University of Bern, Switzerland

C. Pandu Rangan

Indian Institute of Technology, Madras, India

Bernhard Steffen

University of Dortmund, Germany

Madhu Sudan

Massachusetts Institute of Technology, MA, USA

Demetri Terzopoulos

University of California, Los Angeles, CA, USA

Doug Tygar

University of California, Berkeley, CA, USA

Gerhard Weikum

Max-Planck Institute of Computer Science, Saarbruecken, Germany

Colin Fyfe Dongsup Kim Soo-Young Lee
Hujun Yin (Eds.)

Intelligent Data Engineering and Automated Learning – IDEAL 2008

9th International Conference
Daejeon, South Korea, November 2-5, 2008
Proceedings

Volume Editors

Colin Fyfe
University of the West of Scotland
School of Computing
Paisley PA1 2BE, UK
E-mail: colin.fyfe@uws.ac.uk

Dongsup Kim
Soo-Young Lee
Korea Advanced Institute of Science and Technology (KAIST)
Department of Bio and Brain Engineering
Daejeon 305-701, South Korea
E-mail: {kds, sylee}@kaist.ac.kr

Hujun Yin
University of Manchester
School of Electrical and Electronic Engineering
Manchester M60 1QD, UK
E-mail: hujun.yin@manchester.ac.uk

Library of Congress Control Number: 2008937580

CR Subject Classification (1998): H.2.8, H.2, F.2.2, I.2, F.4, K.4.4, H.3, H.4

LNCS Sublibrary: SL 3 – Information Systems and Application, incl. Internet/Web and HCI

ISSN 0302-9743
ISBN-10 3-540-88905-1 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-88905-2 Springer Berlin Heidelberg New York

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer. Violations are liable to prosecution under the German Copyright Law.

Springer is a part of Springer Science+Business Media

springer.com

© Springer-Verlag Berlin Heidelberg 2008
Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 12548484 06/3180 5 4 3 2 1 0

Preface

IDEAL 2008 was the ninth IDEAL conference to take place; earlier editions were held in Hong Kong, the UK, Australia and Spain. This was the first time, though hopefully not the last time, that it took place in Daejeon, South Korea, during November 2–5, 2008.

As the name suggests, the conference attracts researchers who are involved in either data engineering or learning or, increasingly, both. The former topic involves such aspects as data mining (or intelligent knowledge discovery from databases), information retrieval systems, data warehousing, speech/image/video processing, and multimedia data analysis. There has been a traditional strand of data engineering at IDEAL conferences which has been based on financial data management such as fraud detection, portfolio analysis, prediction and so on. This has more recently been joined by a strand devoted to bioinformatics, particularly neuroinformatics and gene expression analysis.

Learning is the other major topic for these conferences and this is addressed by researchers in artificial neural networks, machine learning, evolutionary algorithms, artificial immune systems, ant algorithms, probabilistic modelling, fuzzy systems and agent modelling. The core of all these algorithms is adaptation.

This ninth IDEAL was held in the famous Korea Advanced Institute of Science and Technology (KAIST), in Daejeon, Korea. KAIST is located in Daedeok Science Town, home to more than 60 government-supported and private research institutes, 4 universities, and numerous venture businesses. The Science Town is situated in the northern part of Daejeon, which has a population of over 1.3 million citizens and is obviously an ideal venue for a conference like IDEAL.

The selection of papers was extremely rigorous in order to maintain the high quality of the conference and we would like to thank the members of the Program Committee for their hard work in the reviewing process. This process is essential to the creation of a conference of high standard and the IDEAL conference would not exist without their help.

August 2008

Colin Fyfe
Dongsup Kim
Soo-Young Lee
Hujun Yin

Program Committee Co-chairs

Colin Fyfe
Dongsup Kim
Malik Magdon-Ismael

University of West Scotland, UK
KAIST, Korea
Rensselaer Polytechnic Institute, USA

Program Committee

Ajith Abraham
Khurshid Ahmad
Martyn Amos
Davide Anguita
Javier Bajo
Bruno Baruque
Mikael Boden
Lourdes Borrajo
Juan Botía
Matthew Casey
Darryl Charles
Luonan Chen
Sheng Chen
Shu-Heng Chen
Songcan Chen
Sung-Bae Cho
Seungjin Choi
David A. Clifton
Juan M. Corchado
Rafael Corchuelo
Jose Alfredo F. Costa
Alfredo Cuzzocrea
Sanmay Das
Bernard De Baets
Yanira de Paz
Fernando Díaz
José Dorronsoro
Igor Farkas
Florentino Fernández
Francisco Ferrer
Marcus Gallagher
John Qiang Gan
Mark Girolami
Daniel González
Francisco Herrera
Álvaro Herrero
Michael Herrmann

Jim Hogan
Jaakko Hollmen
David Hoyle
Masumi Ishikawa
Gareth Jones
Ata Kaban
Juha Karhunen
Samuel Kaski
John Keane
Sung-Ho Kim
Mario Koeppen
Pei Ling Lai
Paulo Lisboa
Eva Lorenzo
Frederic Maire
Roque Marín
José F. Martínez
José Ramón Méndez
Simon Miles
Carla Moller-Levet
Yusuke Nojima
Chung-Ming Ou
Jongan Park
Juan Pavón
David Powers
Vic Rayward-Smith
Ramón Rizo
Roberto Ruiz
José Santos
Hyoseop Shin
Michael Small
P.N. Suganthan
Dante Israel Tapia
Peter Tino
Marc M. Van Hulle
Alfredo Vellido
Jose R. Villar

Lipo Wang
Tzai Der Wang
Dong-Qing Wei

Wu Ying
Du Zhang
Rodolfo Zunino

The IDEAL 2008 Organizing Committee would like to acknowledge the financial support of the Department of Bio & Brain Engineering, KAIST and the Air Force Office of Scientific Research, Asian Office of Aerospace Research and Development, USA.

Table of Contents

Learning and Information Processing

Proposal of Exploitation-Oriented Learning PS-r [#]	1
<i>Kazuteru Miyazaki and Shigenobu Kobayashi</i>	
Kernel Regression with a Mahalanobis Metric for Short-Term Traffic Flow Forecasting	9
<i>Shiliang Sun and Qiaona Chen</i>	
Hybrid Weighted Distance Measures and Their Application to Pattern Recognition	17
<i>Zeshui Xu</i>	
A Multitask Learning Approach to Face Recognition Based on Neural Networks	24
<i>Feng Jin and Shiliang Sun</i>	
Logic Synthesis for FSMs Using Quantum Inspired Evolution	32
<i>Marcos Paulo Mello Araujo, Nadia Nedjah, and Luiza de Macedo Mourelle</i>	
A New Adaptive Strategy for Pruning and Adding Hidden Neurons during Training Artificial Neural Networks	40
<i>Md. Monirul Islam, Md. Abdus Sattar, Md. Faijul Amin, and Kazuyuki Murase</i>	
Using Kullback-Leibler Distance in Determining the Classes for the Heart Sound Signal Classification	49
<i>Yong-Joo Chung</i>	
A Semi-fragile Watermark Scheme Based on the Logistic Chaos Sequence and Singular Value Decomposition	57
<i>Jian Li, Bo Su, Shenghong Li, Shilin Wang, and Danhong Yao</i>	
Distribution Feeder Load Balancing Using Support Vector Machines	65
<i>J.A. Jordaan, M.W. Siti, and A.A. Jimoh</i>	
Extracting Auto-Correlation Feature for License Plate Detection Based on AdaBoost	72
<i>Hauchun Tan, Yafeng Deng, and Hao Chen</i>	
Evolutionary Optimization of Union-Based Rule-Antecedent Fuzzy Neural Networks and Its Applications	80
<i>Chang-Wook Han</i>	

Improving AdaBoost Based Face Detection Using Face-Color Preferable Selective Attention	88
<i>Bumhwi Kim, Sang-Woo Ban, and Minho Lee</i>	
Top-Down Object Color Biased Attention Using Growing Fuzzy Topology ART	96
<i>Byungku Hwang, Sang-Woo Ban, and Minho Lee</i>	
A Study on Human Gaze Estimation Using Screen Reflection	104
<i>Nadeem Iqbal and Soo-Young Lee</i>	
A Novel GA-Taguchi-Based Feature Selection Method	112
<i>Cheng-Hong Yang, Chi-Chun Huang, Kuo-Chuan Wu, and Hsin-Yun Chang</i>	
Nonnegative Matrix Factorization (NMF) Based Supervised Feature Selection and Adaptation	120
<i>Pareesh Chandra Barman and Soo-Young Lee</i>	
Automatic Order of Data Points in RE Using Neural Networks	128
<i>Xueming He, Chenggang Li, Yujin Hu, Rong Zhang, Simon X. Yang, and Gauri S. Mittal</i>	
Orthogonal Nonnegative Matrix Factorization: Multiplicative Updates on Stiefel Manifolds	140
<i>Jiho Yoo and Seungjin Choi</i>	
Feature Discovery by Enhancement and Relaxation of Competitive Units	148
<i>Ryotaro Kamimura</i>	
Genetic Feature Selection for Optimal Functional Link Artificial Neural Network in Classification	156
<i>Satchidananda Dehuri, Bijan Bihari Mishra, and Sung-Bae Cho</i>	
A Novel Ensemble Approach for Improving Generalization Ability of Neural Networks	164
<i>Lei Lu, Xiaoqin Zeng, Shengli Wu, and Shuiming Zhong</i>	
Semi-supervised Learning with Ensemble Learning and Graph Sharpening	172
<i>Inae Choi and Hyunjung Shin</i>	
Exploring Topology Preservation of SOMs with a Graph Based Visualization	180
<i>Kadim Taşdemir</i>	
A Class of Novel Kernel Functions	188
<i>Xinfei Liao and Limin Tao</i>	

Data Mining and Information Management

RP-Tree: A Tree Structure to Discover Regular Patterns in Transactional Database	193
<i>Syed Khairuzzaman Tanbeer, Chowdhury Farhan Ahmed, Byeong-Soo Jeong, and Young-Koo Lee</i>	
Extracting Key Entities and Significant Events from Online Daily News	201
<i>Mingrong Liu, Yicen Liu, Liang Xiang, Xing Chen, and Qing Yang</i>	
Performance Evaluation of Intelligent Prediction Models on Smokers' Quitting Behaviour	210
<i>Chang-Joo Yun, Xiaojiang Ding, Susan Bedingfield, Chung-Hsing Yeh, Ron Borland, David Young, Sonja Petrovic-Lazarevic, Ken Coghill, and Jian Ying Zhang</i>	
Range Facial Recognition with the Aid of Eigenface and Morphological Neural Networks	217
<i>Chang-Wook Han</i>	
Modular Bayesian Network Learning for Mobile Life Understanding	225
<i>Keum-Sung Hwang and Sung-Bae Cho</i>	
Skin Pores Detection for Image-Based Skin Analysis	233
<i>Qian Zhang and TaegKeun Whangbo</i>	
An Empirical Research on Extracting Relations from Wikipedia Text	241
<i>Jin-Xia Huang, Pum-Mo Ryu, and Key-Sun Choi</i>	
A Data Perturbation Method by Field Rotation and Binning by Averages Strategy for Privacy Preservation	250
<i>Mohammad Ali Kadampur and Somayajulu D.V.L.N.</i>	
Mining Weighted Frequent Patterns Using Adaptive Weights	258
<i>Chowdhury Farhan Ahmed, Syed Khairuzzaman Tanbeer, Byeong-Soo Jeong, and Young-Koo Lee</i>	
On the Improvement of the Mapping Trustworthiness and Continuity of a Manifold Learning Model	266
<i>Raúl Cruz-Barbosa and Alfredo Vellido</i>	
Guaranteed Network Traffic Demand Prediction Using FARIMA Models	274
<i>Mikhail Dashevskiy and Zhiyuan Luo</i>	
A New Incremental Algorithm for Induction of Multivariate Decision Trees for Large Datasets	282
<i>Anilu Franco-Arcega, J. Ariel Carrasco-Ochoa, Guillermo Sánchez-Díaz, and J. Fco Martínez-Trinidad</i>	

The Use of Semi-parametric Methods for Feature Extraction in Mobile Cellular Networks 290
A.M. Kurien, B.J. Van Wyk, Y. Hamam, and Jaco Jordaan

Personalized Document Summarization Using Non-negative Semantic Feature and Non-negative Semantic Variable 298
Sun Park

Bioinformatics and Neuroinformatics

Cooperative E-Organizations for Distributed Bioinformatics Experiments 306
Andrea Bosin, Nicoletta Dessì, Mariagrazia Fugini, and Barbara Pes

Personal Knowledge Network Reconfiguration Based on Brain Like Function Using Self Type Matching Strategy 314
JeongYon Shim

A Theoretical Derivation of the Kernel Extreme Energy Ratio Method for EEG Feature Extraction 321
Shiliang Sun

Control of a Wheelchair by Motor Imagery in Real Time 330
Kywan Choi and Andrzej Cichocki

Robust Vessel Segmentation Based on Multi-resolution Fuzzy Clustering 338
Gang Yu, Pan Lin, and Shengzhen Cai

Building a Spanish MMTx by Using Automatic Translation and Biomedical Ontologies 346
Francisco Carrero, José Carlos Cortizo, and José María Gómez

Compensation for Speed-of-Processing Effects in EEG-Data Analysis 354
Matthias Ihrke, Hecke Schrobsdorff, and J. Michael Herrmann

Statistical Baselines from Random Matrix Theory 362
Marotessa Voultzidou and J. Michael Herrmann

Adaptive Classification by Hybrid EKF with Truncated Filtering: Brain Computer Interfacing 370
Ji Won Yoon, Stephen J. Roberts, Matthew Dyson, and John Q. Gan

Agents and Distributed Systems

Improving the Relational Evaluation of XML Queries by Means of Path Summaries 378
Sherif Sakr

Identification of the Inverse Dynamics Model: A Multiple Relevance Vector Machines Approach	387
<i>Chuan Li, Xianming Zhang, Shilong Wang, Yutao Dong, and Jing Chen</i>	
When Is Inconsistency Considered Harmful: Temporal Characterization of Knowledge Base Inconsistency	395
<i>Du Zhang and Hong Zhu</i>	
Intelligent Engineering and Its Application in Policy Simulation	404
<i>Xiaoyou Jiao and Zhaoguang Hu</i>	
Design of Directory Facilitator for Agent-Based Service Discovery in Ubiquitous Computing Environments	412
<i>Geon-Ha Lee, Yoe-Jin Yoon, Seung-Hyun Lee, Kee-Hyun Choi, and Dong-Ryeol Shin</i>	
Financial Engineering and Modeling	
Laboratory of Policy Study on Electricity Demand Forecasting by Intelligent Engineering	420
<i>Zhaoguang Hu, Minjie Xu, Baoguo Shan, and Xiandong Tan</i>	
Self-adaptive Mutation Only Genetic Algorithm: An Application on the Optimization of Airport Capacity Utilization	428
<i>King Loong Shiu and K.Y. Szeto</i>	
Cross Checking Rules to Improve Consistency between UML Static Diagram and Dynamic Diagram	436
<i>IlKyu Ha and Byunguk Kang</i>	
Neural Networks Approach to the Detection of Weekly Seasonality in Stock Trading	444
<i>Virgilijus Sakalauskas and Dalia Kriksciuniene</i>	
Invited Session	
Bregman Divergences and the Self Organising Map	452
<i>Eunsong Jang, Colin Fyfe, and Hanseok Ko</i>	
Feature Locations in Images	459
<i>Hokun Kim, Colin Fyfe, and Hanseok Ko</i>	
A Hierarchical Self-organised Classification of ‘Multinational’ Corporations	464
<i>Khurshid Ahmad, Chaoxin Zheng, and Colm Kearney</i>	

An Adaptive Image Watermarking Scheme Using Non-separable Wavelets and Support Vector Regression	473
<i>Liang Du, Xinge You, and Yiu-ming Cheung</i>	
Cluster Analysis of Land-Cover Images Using Automatically Segmented SOMs with Textural Information	483
<i>Márcio L. Gonçalves, Márcio L.A. Netto, and José A.F. Costa</i>	
Application of Topology Preserving Ensembles for Sensory Assessment in the Food Industry	491
<i>Bruno Baruque, Emilio Corchado, Jordi Rovira, and Javier González</i>	
AI for Modelling the Laser Milling of Copper Components	498
<i>Andrés Bustillo, Javier Sedano, José Ramón Villar, Leticia Curiel, and Emilio Corchado</i>	
Country and Political Risk Analysis of Spanish Multinational Enterprises Using Exploratory Projection Pursuit	508
<i>Alfredo Jiménez, Álvaro Herrero, and Emilio Corchado</i>	
Single-Layer Neural Net Competes with Multi-layer Neural Net	516
<i>Zheng Rong Yang</i>	
Semi-supervised Growing Neural Gas for Face Recognition	525
<i>Shireen Mohd Zaki and Hujun Yin</i>	
Author Index	533

Proposal of Exploitation-Oriented Learning PS-r[#]

Kazuteru Miyazaki¹ and Shigenobu Kobayashi²

¹ National Institution for Academic Degrees and University Evaluation,
1-29-1, Gakuennishimachi, Kodaira-city, Tokyo 187-8587, Japan,
teru@niad.ac.jp,

<http://svrrd2.niad.ac.jp/faculty/teru/index.html>

² Tokyo Institute of Technology
4259, Nagatsuta, Midori-ku, Yokohama, Kanagawa, 226-8502, Japan

Abstract. *Exploitation-oriented Learning* (XoL) is a novel approach to goal-directed learning from interaction. Though *reinforcement learning* is much more focus on the learning and can guarantee the optimality in *Markov Decision Processes* (MDPs) environments, XoL aims to learn *a rational policy*, whose expected reward per an action is larger than zero, very quickly. We know PS-r* that is one of the XoL methods. It can learn *an useful rational policy* that is not inferior to a random walk in *Partially Observed Markov Decision Processes* (POMDPs) environments where the number of types of a reward is one. However, PS-r* requires $O(MN^2)$ memories where N and M are the numbers of types of a sensory input and an action. In this paper, we propose PS-r[#] that can learn an useful rational policy in the POMDPs environments by $O(MN)$ memories. We confirm the effectiveness of PS-r[#] in numerical examples.

1 Introduction

The approach, called *reinforcement learning* (RL), is much more focused on goal-directed learning from interaction than are other approaches to machine learning [12]. It is very attractive since it can use *Dynamic Programming* (DP) to analyze its behavior. We call these method that is based on DP *DP-based RL method*. In general, RL uses *a reward* as a teacher signal for its learning. The DP-based RL method aims to optimize its behavior under the values of reward signals that are designed by RL users.

We want to apply RL to many real world problems more easily. Though we know some important applications [4], generally speaking, it is difficult to design RL systems to fit on a real world problem. We think that the following two reasons concern with it. In the first, the interaction will require many trial-and-error searches. In the second, there is no guideline how to design the values of reward signals. Though they are not treated as important issues on theoretical papers, they are able to be a serious issue in real world applications. Especially, if we have assigned inappropriate values to reward signals, we will receive an unexpected result [8].

We know the *Inverse Reinforcement Learning* (IRL) [11,12] as a method related to the design problem of the values of reward signals. If we input our expected policy to the IRL system, it can output a *reward function* that can realize the policy. On the other hand, we are interested in the approach where reward signals are treated independently and do not assign any value to them. Furthermore, we aim to reduce the number of trial-and-error searches through strongly enhancing successful experiences. We call it *Exploitation-oriented Learning* (XoL). As examples of learning systems that can belong in XoL, we know the rationality theorem of *Profit Sharing* (PS) [6], the *Rational Policy Making algorithm* [7], the *Penalty Avoiding Rational Policy Making algorithm* [8] and *PS-r** [9].

XoL has several features. (1) Though traditional RL systems require appropriate values of reward signals, XoL only requires an order of importance among them. In general, it is easier than designing their values. (2) XoL can learn more quickly since it traces successful experiences very strongly. (3) XoL may be unsuitable for pursuing the optimality. It can be guaranteed by the *multi-start method* [7] that resets all memories to get a better policy. (4) XoL is effective on the classes beyond MDPs, since it is a *Bellman-free method* [12].

In this paper, we aim to improve PS-r*. PS-r* can learn an *useful rational policy* that is not inferior to a random walk in *Partially Observed Markov Decision Processes* (POMDPs) environments where the number of types of a reward is one, whereas a DP-based RL method cannot always learn it [9]. However, PS-r* requires $O(MN^2)$ memories where N and M are the number of types of a sensory input and an action. In this paper, we propose PS-r# that can learn an useful rational policy in the POMDPs environments by $O(MN)$ memories. We confirm the effectiveness of PS-r# in numerical examples.

2 The Domain

2.1 Notations

Consider an agent in some unknown environment. The agent senses a set of discrete attribute-value pairs and performs an action in some discrete varieties. The environment provides a reward signal to the agent as a result of some sequence of action. A sensory input and an action constitute a pair that is termed as a *rule*. The function that maps sensory inputs to actions is called a *policy*. We call a policy *rational* if and only if the expected reward per an action is larger than zero. Furthermore, an *useful rational policy* is a rational policy that is not inferior to the *random walk* (RA) where the agent selects an action based on the same probability to every action in every sensory input.

We term the sequence of rules selected between rewards as an *episode*. We term the subsequent episode as a *detour* when the sensory input of the first selection rule and the sensory output of the last selection rule are the same although both rules are different. The detour is also called an *unrewarded loop*. The rules on a detour may not contribute to obtain a reward. We term a rule as *irrational* if and only if it always exists on detours in any episode. Otherwise, a rule is

termed as *rational*. An irrational rule should not be selected when they conflict with a rational rule. Examples of these terms are shown in papers [6,7,8,9,10].

2.2 Properties of the Target Environments

We focus on the POMDPs environments where the number of types of a reward is one. We understand that POMDP is a class that is representative of non-Markovian environments. In POMDPs, the agent senses different environmental states as the same sensory input. We call the sensory input a *Partially Observable* (PO) sensory input.

We recognize that the complete implementation of POMDPs must overcome two deceptive problems [7]. We term the indistinguishable of state values as a

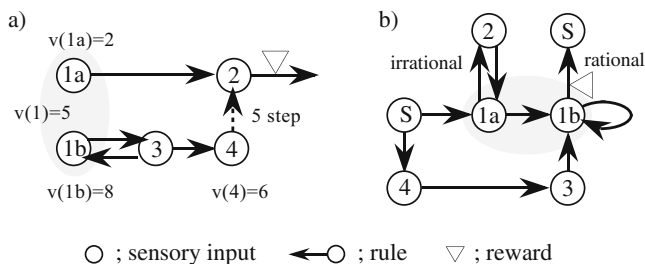


Fig. 1. Examples of type 1(a) and type 2(b) confusions

type 1 confusion. Fig. 1a is an example of the type 1 confusion. In this example, the state value (v) is estimated by the minimum number of steps required to obtain a reward [7]. The values for the states 1a and 1b are 2 and 8, respectively. Although the state 1a and 1b are different states, the agent senses them as the same sensory input (a state 1). If the agent experiences the state 1a and 1b equally likely, the value of the state 1 becomes $5 (= \frac{2+8}{2})$. Therefore, the value of the state 1 is higher than that of the state 4 (that is 6). If the agent uses the state values for its learning, it would prefer to move *left* when in the state 3. However, the agent has to move *right* in the state 1. This implies that the agent has learned the irrational policy, where it only transits between the states 1b and 3.

We term the indistinguishable of rational and irrational rules as a *type 2 confusion*. Fig. 1b is an example of the type 2 confusion. Although the action of moving *up* in the state 1a is irrational, it is rational in the state 1b. Since the agent senses the states 1a and 1b as the same sensory input (the state 1), the action of moving *up* in the state 1 is regarded as rational. If the agent learns the action of moving *right* in the state S, it will fall into an irrational policy where the agent only transits between the states 1a and 2.

In general, if there exists a type 2 confusion in some sensory input, there exists a type 1 confusion in it. Q-learning (QL), that is a representative DP-based RL

¹ Remark that the highest state value is 1 that is assigned in state 2.

method, is deceived by a type 1 confusion, since it uses state values to formulate a policy. As the XoL methods do not use state values, they are not deceived by the confusion. On the other hand, the type 2 confusion is more difficult to treat than the type 1 confusion, in general. QL and the XoL methods except PS-r* have possibility to be deceived by the type 2 confusion.

3 Proposal of the PS-r# Algorithm

3.1 Traditional Approaches to POMDPs

To treat the POMDPs environments, especially the type 2 confusions, many methods are proposed. The most traditional approach is *the memory-based approach* ([2,5] and so on.) that uses a series of sensor-action pairs or model to identify a PO sensory input. Although the memory-based approach can attain optimality, it is difficult to apply to the case of many state-action spaces, for example, hundreds of thousands or over a million state-action spaces.

To resolve the problem using the memory-based approach, *a stochastic policy* is proposed, where the agent selects an action based on the non-zero probability of every action in every sensory input in order to escape a PO sensory input. The most simplest stochastic policy is the random walk (RA) that assigns the same probability to every action. On the other hand, the existing RL systems to learn a stochastic policy ([13,3] and so on.) are types of hill-climbing methods. They are often used in the POMDP environments, since they can converge to *a local optimum*. However, they cannot always improve RA. Furthermore, we know a case where they change for a policy worse than RA [9].

3.2 Our Approach to POMDPs

To avoid the fault with the hill-climbing methods, we focus on XoL that is a non-hill-climbing approach. We know the rationality theorem of PS in a subclass of the POMDPs where there is no type 2 confusion [7]. The properties of PS in the class has been analyzed through *PS-r* [9] that is an abstract algorithm of PS.

PS-r uses a memory called *1st memory* to evaluate all rules. At the beginning of learning, all rules are regarded as irrational and are initialized by r ($0 < r < 1$) points. If a rule is regarded as rational, r of the rule is updated to 1, which means that is a rational rule. Once r is set to 1, this value is maintained throughout. While learning is in progress, an action is selected based on RA. Therefore, PS-r can find all rational rules. When we have evaluated or utilized a policy that is learned by PS-r, an action is selected based on a roulette selection in proportion to r .

Furthermore, we know *PS-r** [9], that is an extended algorithm of PS-r that has been modified to fit for the POMDPs environments where there is a type 2 confusion. PS-r* can learn an useful rational policy in the POMDPs environments where the number of types of a reward is one. However, it requires $O(MN^2)$ memories where N and M are the numbers of types of a sensory input and an action, since it uses χ^2 -goodness fit test to find a PO sensory input.

3.3 The PS-r[#] Algorithm

We aim to reduce the memory to $O(MN)$ by proposing the PS-r[#] algorithm. PS-r[#] is based on PS-r. If we have a policy that is learned by PS-r and it is an useful rational policy, we do not change the policy. On the other hand, in general, the policy that is learned by PS-r may not be an useful rational policy. Especially, if rewards cannot be obtained after *a certain number of actions*, for example, X times compared to when a reward is obtained at first, the policy will not be an useful rational policy. In this case, we regard the agent as falling into an unrewarded loop. In the unrewarded loop, the action is selected by RA at each sensory input. By this process, we can get a reward in the case that has a PO sensory input.

The unrewarded loop can easily be determined by memorizing an episode that is used by some XoL methods [6,7,8,9,10] in general. For example, after the agent senses $S_2, S_1, S_3, S_2, S_1, S_0, S_4$ and S_5 sensory inputs in this order, if the number of actions will be larger than the certain number of actions, we can regard (S_3, S_2, S_1) as an unrewarded loop.

PS-r[#] do not use χ^2 -goodness fit test to find a PO sensory input. Also, PS-r[#] is not inferior to RA since it uses RA in an unrewarded loop. Therefore, it can get an useful rational policy by $O(MN)$ memories in the same POMDPs environments that is treated with PS-r*

4 Evaluation of the PS-r[#] Algorithm

4.1 Comparison with PS-r* and SGA

PS-r[#] is compared with PS-r* and SGA [3], that can learn a stochastic policy in POMDPs, in the environment that is shown in Fig.2. In this environment, the different environmental states Z_a, Z_b, Z_c , and Z_d are sensed as the same sensory input Z . After the agent selects action-a in sensory input X , it moves to the sensory inputs S_1 and X with p and $1 - p$ probabilities, respectively. States from S_1 to S_n are sensed as n different sensory inputs. If we adjust n and p , the average number of steps required to obtain a reward can be changed.

The learning parameter and discounted rate of SGA are 0.1 and 0.99, respectively. The initial policy given to SGA is RA. In PS-r*, for χ^2 -goodness fit test, we have to set *a significant level*, *a detection power* and the maximum error of estimation of transition probabilities [2]. In this paper, we have set these parameters as 0.05, 0.90 and 0.05, respectively.

We carried out 100 trials with different random seeds. We have changed n to 7, 14 and 21. Table 1 shows *the quality* (QUA.) that is the average number of steps required to obtain a reward and *the speed* (SPD.) that is the average number of steps required to reach the quality.

PS-r[#] has parameter X that is used to find an unrewarded loop. If rewards cannot be obtained after the number of actions that is determined by X times

² Details of these parameters are shown in the paper [9].

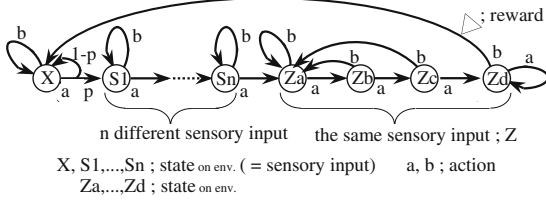


Fig. 2. The environment in which PS-r[#] is compared with PS-r* and SGA

Table 1. The results of comparison with PS-r[#], PS-r* and SGA in Fig.2

n	PS-r [#] ($X \geq 1$)		PS-r*		SGA	
	QUA.	SPD.	QUA.	SPD.	QUA.	SPD.
7	24.0	$32.4 * (X + 1)$	24.2	$2.38 * 10^4$	26.3	$2.21 * 10^3$
14	30.1	$44.3 * (X + 1)$	31.2	$1.73 * 10^5$	38.0	$4.27 * 10^3$
21	38.2	$59.3 * (X + 1)$	38.2	$2.19 * 10^5$	50.8	$4.42 * 10^3$



Fig. 3. The maze environment

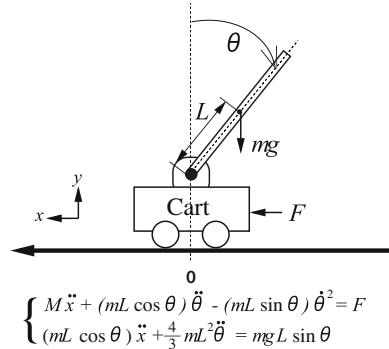


Fig. 4. The pole-cart system

compared to when a reward is obtained at first, we regard the agent as falling into an unrewarded loop.

In this experiment, if we set $X \geq 1$, PS-r[#] have get the quality that is shown in table 1. We have confirmed that PS-r[#] can get the same quality with PS-r* and more faster than PS-r*.

4.2 Evaluation in a Maze and a Pole-Cart Problems

Next, PS-r[#] is compared with PS-r* in the maze environment that is shown in Fig.3, and SGA in the pole-cart system that is shown in Fig.4. We carried out 100 trials with different random seeds in each problem. We set $X = 1$.

In the maze environment, we have changed r to 0.8, 0.5 and 0.1. Table 2 shows the quality (QUA.) that is the average number of steps to reach to the goal G

Table 2. The results of comparison with PS-r[#] and PS-r* in Fig.3

	r	QUA.	SPD.
PS-r [#]	0.8	22.7	4.75
	0.5	22.7	4.75
	0.1	22.7	4.75
PS-r*	0.8	36.7	$5.24 * 10^3$
	0.5	35.3	$7.13 * 10^5$
	0.1	36.7	$6.02 * 10^6$

Table 3. The results of comparison with PS-r[#] and SGA in Fig.4

	d	No. of rewards getting			
		1	2	10	100
PS-r [#]	3	715.4	66.0	58.9	52.3
	5	822.6	66.5	50.0	48.7
	7	849.2	70.6	50.1	48.5
SGA	3	785.6	820.6	533.4	69.1
	5	669.2	651.3	562.5	124.0
	7	869.6	820.8	765.3	262.8

from S_0 and the speed (SPD.) that is the average number of steps required to reach the quality. Though the performance of PS-r* has been influenced by r , PS-r[#] is not influenced by it. We have confirmed that PS-r[#] can get more better quality than PS-r* and more faster than PS-r*.

In the pole-cart system, m is the mass of a pendulum ($m = 0.1kg$), M being the sum of masses of the pendulum and cart ($M = 1.1kg$), $2L$ being the length of the pendulum ($2L = 1.0m$), and F being force exerted on the cart ($F = -40.0, 0.0, or, 40.0$). An initial position is a state, in which the pendulum stands still right below and the cart at the center. Four-dimensional continuous values are given for sensory input: { the location of the cart (x), velocity of the cart (\dot{x}), angle of the pendulum(θ), and angular velocity of the pendulum ($\dot{\theta}$) }. For digitizing, these continuous values are equally divided by parameter d ($d = 3, 5, 7$). The range of motion of the cart is $-2.4 < x < 2.4$. Beyond this range, the cart is returned to the initial position. When the pendulum is raised, the agent can get a reward.

Table 3 shows the number of actions to get a reward on 1, 2, 10 and 100 rewards getting for each d . We have confirmed that PS-r[#] is very quickly than SGA. Furthermore, we applied PS-r[#] to parallel double inverted pendulums. If we set $d = 9$, it has 1,594,323 state-action spaces. In this case, SGA and memory-based approaches could not find an useful rational policy. On the other hand, PS-r[#] can get a reward by 5057.2, 73.7, 72.0 and 53.7 actions on 1, 2, 10 and 100 rewards getting, respectively. It means that PS-r[#] can find an useful rational policy very quickly in the case of a million state-action spaces.

5 Conclusions

In this paper, we have proposed *Exploitation-oriented Learning* (XoL) that is a novel approach to goal-directed learning from interaction. XoL can learn a rational policy very quickly and does not require a value of a reward signal. We know PS-r* that is one of the XoL methods and can learn *an useful rational policy* that is not inferior to a random walk in POMDPs environments where the number of types of a reward is one. However, PS-r* requires $O(MN^2)$ memories where N and M are the numbers of types of a sensory input and an action.

In this paper, we have proposed PS-r[#] that can learn an useful rational policy in the POMDPs environments by $O(MN)$ memories. We have confirmed the effectiveness of PS-r[#] in numerical examples.

In the future, we will extend PS-r[#] to the environments that have continuous state spaces [10] and have several reward signals at the same time [8]. Furthermore, we will find an efficient application as soon as possible.

References

1. Abbeel, P., Ng, A.Y.: Exploration and apprenticeship learning in reinforcement learning. In: Proc. of 22th International Conference on Machine Learning, pp. 1–8 (2005)
2. Chrisman, L.: Reinforcement Learning with perceptual aliasing: The Perceptual Distinctions Approach. In: Proc. of 10th National Conference on Artificial Intelligence, pp. 183–188 (1992)
3. Kimura, H., Yamamura, M., Kobayashi, S.: Reinforcement Learning by Stochastic Hill Climbing on Discounted Reward. In: Proc. of 12th International Conference on Machine Learning, pp. 295–303 (1995)
4. Merrick, K., Maher, M.L.: Motivated Reinforcement Learning for Adaptive Characters in Open-Ended Simulation Games. In: Proc. of the International Conference on Advanced in Computer Entertainment Technology, pp. 127–134 (2007)
5. McCallum, R.A.: Instance-Based Utile Distinctions for Reinforcement Learning with Hidden State. In: Proc. of 12th International Conference on Machine Learning, pp. 387–395 (1995)
6. Miyazaki, K., Yamamura, M., Kobayashi, S.: On the Rationality of Profit Sharing in Reinforcement Learning. In: Proc. of 3rd International Conference on Fuzzy Logic, Neural Nets and Soft Computing, pp. 285–288 (1994)
7. Miyazaki, K., Kobayashi, S.: Learning Deterministic Policies in Partially Observable Markov Decision Processes. In: Proc. of 5th International Conference on Intelligent Autonomous System, pp. 250–257 (1998)
8. Miyazaki, K., Kobayashi, S.: Reinforcement Learning for Penalty Avoiding Policy Making. In: Proc. of the 2000 IEEE International Conference on Systems, Man and Cybernetics, pp. 206–211 (2000)
9. Miyazaki, K., Kobayashi, S.: An Extension of Profit Sharing to Partially Observable Markov Decision Processes: Proposition of PS-r* and its Evaluation. Journal of the Japanese Society for Artificial Intelligence 18(5), 286–296 (2003) (in Japanese)
10. Miyazaki, K., Kobayashi, S.: A Reinforcement Learning System for Penalty Avoiding in Continuous State Spaces. Journal of Advanced Computational Intelligence and Intelligent Informatics 11(6), 668–676 (2007)
11. Ng, A.Y., Russell, S.J.: Algorithms for Inverse Reinforcement Learning. In: Proc. of 17th International Conference on Machine Learning, pp. 663–670 (2000)
12. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction, A Bradford Book. MIT Press, Cambridge (1998)
13. Williams, R.J.: Simple Statistical Gradient Following Algorithms for Connectionist Reinforcement Learning. Machine Learning 8, 229–256 (1992)

Kernel Regression with a Mahalanobis Metric for Short-Term Traffic Flow Forecasting

Shiliang Sun and Qiaona Chen

Department of Computer Science and Technology,
East China Normal University, Shanghai 200241, China
s1sun@cs.ecnu.edu.cn

Abstract. In this paper, we apply a new method to forecast short-term traffic flows. It is kernel regression based on a Mahalanobis metric whose parameters are estimated by gradient descent methods. Based on the analysis for eigenvalues of learned metric matrices, we further propose a method for evaluating the effectiveness of the learned metrics. Experiments on real data of urban vehicular traffic flows are performed. Comparisons with traditional kernel regression with the Euclidean metric under two criterions show that the new method is more effective for short-term traffic flow forecasting.

Keywords: traffic flow forecasting, kernel regression, Mahalanobis metric, Euclidean metric, gradient descent.

1 Introduction

Traffic flow forecasting, which contributes a lot to traffic signal control and congestion avoidance, plays an important role in the research area of intelligent transportation systems (ITS). Recently, increasing worldwide attentions are being attracted to this topic. Short-term traffic flow forecasting, which aims to predict traffic flows in the near future, usually five to thirty minutes later, is the focus of this paper.

Till now, there are some methods proposed for short-term traffic flow forecasting such as time series models [1,2,3], neural network approaches [4], Kalman filter theory [5], Markov chain models [6], support vector machines [7], and Bayesian networks [8]. In addition, kernel regression [9,10,11], our concern in this paper, is also a classical method for traffic flow forecasting.

Although kernel regression is often used, from a careful review we still find a problem. In traditional kernel regression, the Euclidean metric is adopted to calculate weights of different past flows on their contributions to the flow to be forecasted. However, there is a defect about the Euclidean metric for kernel regression. Basically, every past traffic flow should take a different position in the constructed prediction models. That is, some are closely related to future flows, while others may be unrelated to them at all. And this relationship had better be determined by the unique characteristic of every considered data set.

In other words, the relationship should be learned from data. But, the Euclidean metric for kernel regression ignores this and treats it equally by a same metric.

Metric learning for kernel regression proposed by Weinberger and Tesauro [12] combines kernel regression and Mahalanobis metrics together, using gradient descent to minimize the training error. It has the potential to overcome the above mentioned drawback of the Euclidean metric for kernel regression. In this paper, the idea of learning a Mahalanobis metric for kernel regression is transferred to the problem of short-term traffic flow forecasting. In addition, we also present a method for evaluating the effectiveness of learned metrics in terms of eigenvalues. Experimental results with real data indicate that for traffic flow forecasting Mahalanobis metric learning can improve kernel regression consistently.

The rest of this paper is organized as follows. Section 2 introduces kernel regression and metric learning for our concerned traffic flow forecasting problem. Besides reporting experimental results, Section 3 also includes exception analysis where we present the method of evaluating the effectiveness of learned metrics. Finally, Section 4 concludes the paper and gives future research directions.

2 Regression Model for Traffic Flow Forecasting

2.1 A General Regression Model

Given training pattern pairs $(\vec{x}_1, y_1), (\vec{x}_2, y_2), (\vec{x}_3, y_3), \dots, (\vec{x}_n, y_n) \in R^d \times R$, standard regression is to find out $\hat{y}_i = g(\cdot)$ as an estimation of y_i , where g models the forecasting relationship $(y_i = g(\cdot) + \varepsilon)$, at the lowest loss [12]:

$$L = \sum_i (y_i - \hat{y}_i)^2. \quad (1)$$

In a real traffic network, generally current traffic flows are related to flows of past time. If we treat this relationship as a linear one, the current flow on a certain spot can be forecasted using its previous flows as:

$$\hat{y}_i = a_1 \cdot y_{i-1} + a_2 \cdot y_{i-2} + \dots + a_m \cdot y_{i-m}, \quad (2)$$

where \hat{y}_i is the predicted traffic flow, and $y_{i-1}, y_{i-2}, \dots, y_{i-m}$ are flows of past time. The goal of this paper is to estimate the parameters a_1, a_2, \dots, a_m as accurately as possible. Usually, the weighted average method can be used to get an estimation of a_1, a_2, \dots, a_m in which case the above equation is expressed as follows:

$$\hat{y}_i = \frac{y_{i-1} \cdot w_1 + y_{i-2} \cdot w_2 + \dots + y_{i-m} \cdot w_m}{\sum_{j=1}^m w_j}, \quad (3)$$

where w_j is a weight determined by the similarity of y_j ($j = i - 1, \dots, i - m$) to the current value y_i based on some measurement. A larger similarity means a larger weight and a greater impact.

2.2 Kernel Regression

In kernel regression, the distance between the first elements of pattern pairs is defined to measure the similarity between the second elements. A short distance stands for a high similarity and a large kernel value. In terms of kernel terminology, we can rewrite (3) as :

$$\hat{y}_i = \frac{\sum_{j=i-m}^{i-1} y_j \cdot k_{ij}}{\sum_{j=i-m}^{i-1} k_{ij}}, \quad (4)$$

where $k_{ij} = k(\vec{x}_i, \vec{x}_j)$ is a kernel function between \vec{x}_i and \vec{x}_j from the pattern pairs (\vec{x}_i, y_i) and (\vec{x}_j, y_j) . As in [12], in this paper a Gaussian kernel is adopted with the following form:

$$k_{ij} = \frac{1}{\sigma\sqrt{2\pi}} \cdot \exp^{-\frac{d(\vec{x}_i, \vec{x}_j)}{\sigma^2}}, \quad (5)$$

where $d(\vec{x}_i, \vec{x}_j)$ is the squared distance between the vectors \vec{x}_i and \vec{x}_j . For simplicity, σ is often fixed to be 1.

Traditional kernel regression uses the Euclidean metric to calculate $d(\vec{x}_i, \vec{x}_j)$. As stated previously, the metric in kernel regression should adapt with respect to different data sets in order to discover different relations. Therefore, in this paper a Mahalanobis metric is adopted to replace the Euclidean metric in evaluating the distance $d(\vec{x}_i, \vec{x}_j)$. By studying the structure embedded in historical data, it can put more weights on desirable factors, and thus is promising to lead to more accurate regression models.

2.3 Mahalanobis Metric Learning

A Mahalanobis metric is defined as :

$$d(\vec{x}_i, \vec{x}_j) = (\vec{x}_i - \vec{x}_j)^\top M (\vec{x}_i - \vec{x}_j), \quad (6)$$

where metric matrix M can be any symmetric positive semidefinite real matrix [12]. Setting M to be the identity matrix can recover the standard Euclidean metric. Mathematically, M can be decomposed as $M = A^\top A$. Now (6) can be reformulated as:

$$d(\vec{x}_i, \vec{x}_j) = \|A(\vec{x}_i - \vec{x}_j)\|^2. \quad (7)$$

Gradient descent [13] is a well-known method for minimizing loss on training data, and here is employed to carry out metric learning. Gradient descent steps for Mahalanobis metric learning can be described as follows:

- *begin.* initialize : A , stopping criterion θ , learning rate $\eta(k)$, $k = 0$
- *do*
- $\nabla L(A) \leftarrow \frac{\partial L}{\partial A}$, $A \leftarrow A - \eta(k)\nabla L(A)$, $k \leftarrow k + 1$

- until $\eta(k)\nabla L(A) < \theta$
- return A
- end.

Consulting the derivation of $\nabla L(A)$ given in [12], here we provide its derivation for our current problem settings as follows.

$$\frac{\partial L}{\partial A} = -2 \sum_i (y_i - \hat{y}_i) \frac{\partial \hat{y}_i}{\partial A}, \quad (8)$$

$$\frac{\partial \hat{y}_i}{\partial A} = \frac{\left(\sum_{j=i-m}^{i-1} y_j \frac{\partial k_{ij}}{\partial A} \right) \sum_{j=i-m}^{i-1} k_{ij} - \left(\sum_{j=i-m}^{i-1} y_j k_{ij} \right) \sum_{j=i-m}^{i-1} \frac{\partial k_{ij}}{\partial A}}{\left(\sum_{j=i-m}^{i-1} k_{ij} \right)^2}, \quad (9)$$

$$\frac{\partial k_{ij}}{\partial A} = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{d(\vec{x}_i, \vec{x}_j)}{\sigma^2}} \left(-\frac{1}{\sigma^2} \right) \frac{\partial d(\vec{x}_i, \vec{x}_j)}{\partial A}, \quad (10)$$

$$\frac{\partial d(\vec{x}_i, \vec{x}_j)}{\partial A} = 2A(\vec{x}_i - \vec{x}_j)(\vec{x}_i - \vec{x}_j)^T. \quad (11)$$

Rewrite (8) with (9), (10), (11) and $\sigma = 1$, we have

$$\nabla L(A) = \frac{\partial L}{\partial A} = 4A \sum_i (\hat{y}_i - y_i) \frac{\sum_{j=i-m}^{i-1} (\hat{y}_i - y_j) k_{ij} (\vec{x}_i - \vec{x}_j)(\vec{x}_i - \vec{x}_j)^T}{\sum_{j=i-m}^{i-1} k_{ij}}. \quad (12)$$

3 Experiment

3.1 Data Description and Configuration

The data used in our experiments are from Beijing’s Traffic Management Bureau. Fig. 1 is a patch of urban traffic map of highways where traffic flows are recorded. Each circle in the sketch map denotes a road junction, and an arrow shows the direction of the corresponding traffic flow [8]. Vehicular flow rates of discrete time series are recorded every 15 minutes. The recording period is 25 days (totally 2400 sample points) from March, 2002. In the experiments, samples are divided into two sets, one (2112 points from the first 22 days) for training, the other (the rest points) for algorithm test. For evaluation, experiments are performed with multiple randomly selected roads from Fig. 1.

Let $x_1, x_2, x_3, \dots, x_{2400}$ be the original 2400 samples, from which we need first to form the pattern pairs required by our former formulation in Section 2. For a pattern pair (\vec{x}_i, y_i) , \vec{x}_i is used to measure the similarity of y_i with other counterparts. For the current traffic flow forecasting problem, the representation of y_i is direct, that is, x_i , while \vec{x}_i is constructed by d values $x_{i-d}, x_{i-d+1}, \dots, x_{i-1}$.

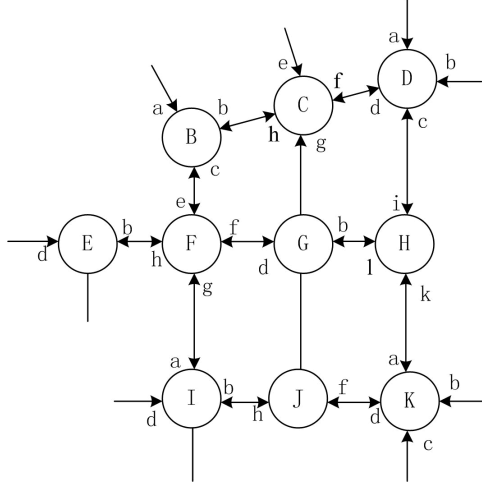


Fig. 1. A patch of traffic map taken from the East Section of the Third Circle of Beijing City Map where the UTC/SCOOT system is mounted [\[8\]](#)

Then $y_i, y_{i+1}, \dots, y_{i+m-1}$ are used to predict y_{i+m} . If number m is large, forecasting results tend to be more accurate. But, it is at the cost of computing more kernel values. In addition, because A is a $d \times d$ matrix, d is also closely related to running time. After bearing a balance between running time and accuracy, we empirically fix d as 10, and m as 5.

3.2 Result

Two widely-used criterions, namely root mean square error (RMSE) and mean absolute relative error (MARE), are adopted to evaluate experimental results. They have the following formulations: $RMSE = \sqrt{\frac{1}{T} \sum_{i=1}^T (y_i - \hat{y}_i)^2}$, $MARE = \frac{1}{T} \sum_{i=1}^T \frac{|y_i - \hat{y}_i|}{y_i}$ with T being the length of series to be forecasted.

Table 1 presents forecasting errors for models with and without metric learning (ML) on training sets. Table 2 presents forecasting errors for these two models on test sets. Symbols Ba , Cf , Fe , Gb , Hi denote the chosen roads for the experiments. The first letter represents the road junction, and the second one represents the link whose stream comes towards the junction. The term ratio is defined as:

$$ratio = \frac{error_{NoML} - error_{ML}}{error_{NoML}}.$$

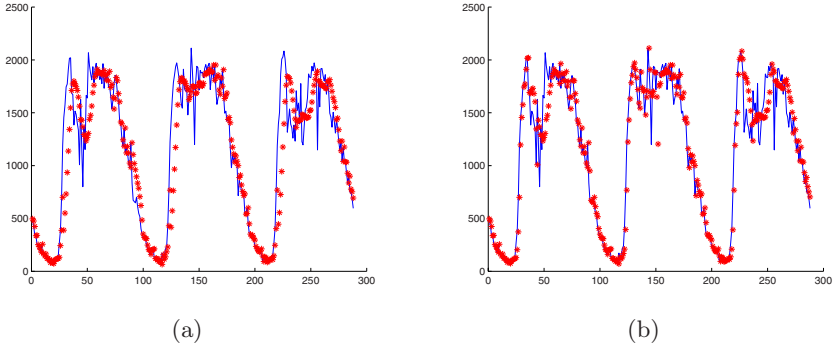
Positive ratios mean error decreasing with the help of metric learning. From Table 1 and Table 2, we can find out that all the ratios on training and test sets are positive except for Cf . Thus, we can draw a cursory conclusion: metric learning improves the learning ability of kernel regression as well as its generalizing ability. We will give an exception analysis on the forecasting performance for Cf shortly.

Table 1. Training error comparison

ML	RMSE			MARE		
	No	Yes	ratio	No	Yes	ratio
Ba	196.2	150.7	23.19%	0.155	0.146	5.81%
Cf	107.3	105.5	1.68%	0.117	0.118	-0.86%
Fe	232.9	155.5	33.23%	0.124	0.114	8.07%
Gb	91.2	85.1	6.69%	0.162	0.154	4.94%
Hi	98.9	90.3	8.70%	0.173	0.167	3.47%

Table 2. Test error comparison

ML	RMSE			MARE		
	No	Yes	ratio	No	Yes	ratio
Ba	237.4	195.5	17.65%	0.165	0.150	9.09%
Cf	112.9	108.9	3.54%	0.105	0.107	-1.91%
Fe	258.5	166.4	35.63%	0.126	0.109	13.49%
Gb	108.7	104.3	4.05%	0.160	0.153	4.38%
Hi	114.2	105.0	8.06%	0.164	0.156	4.88%

**Fig. 2.** Forecasting results of kernel regression without (a) and with (b) metric learning for Ba

Further to give an intuitive illustration of the forecasting performance, we draw the forecasting results of Ba on the test set without and with metric learning respectively, as shown in Fig. 2, where blue lines represent real recorded data and red stars represent forecasted results.

3.3 Exception Analysis

Now we give an analysis on why the forecasting performance of metric learning for Cf is different. In Section 2, it has been mentioned that metric matrix M is symmetric and positive semidefinite, which means that M can be diagonalized,

Table 3. Eigenvalues of learned metric matrices

λ	Num									
	1	2	3	4	5	6	7	8	9	10
<i>Ba</i>	0.4631	0.4834	0.4860	0.4879	0.4882	0.4897	0.4905	0.4917	0.5055	0.5424
<i>Cf</i>	0.3600	0.3600	0.3600	0.3600	0.3600	0.3600	0.3600	0.3600	0.3600	0.3600
<i>Fe</i>	0.1960	0.2136	0.2260	0.2323	0.2375	0.2456	0.2551	0.2731	0.2782	0.3472
<i>Gb</i>	0.1358	0.1831	0.2898	0.3762	0.5011	0.5345	0.5692	0.7744	1.3137	1.5186
<i>Hi</i>	0.0437	0.0777	0.0896	0.1360	0.1738	0.2195	0.3324	0.5707	0.6920	1.5251

Table 4. Training and test error comparison for *Cf*

	RMSE			MARE		
	ML	No	Yes	ratio	No	Yes
Training	107.3	105.4	1.82%	0.117	0.116	1.20%
Test	112.9	109.5	3.05%	0.105	0.104	0.38%

for example, as $M = UAU^T$ with A being the diagonal eigenvalue matrix and U being the corresponding eigenvector matrix. Further considering that $M = A^T A$, we know that A is closely related to the influence of different input factors [12]. Generally, A should have different diagonal elements to account for different influences from different past traffic flows.

Table 3 lists eigenvalues of matrix M obtained in our experiments, from which we can find that eigenvalues in a row are mutually different except for *Cf*. In other words, all the eigenvalues of learned metric matrix M for *Cf* are the same. This accounts for the exception of its forecasting error mode. The abnormality of eigenvalues can be explained by gradient decent methods, which always limits to a local minimum. Therefore, we can automatically select initial values by investigating the eigenvalues of the learned matrix M . After reselecting initial values, we perform this experiment again hopefully to obtain a good result. Table 4 shows the training and forecasting result for *Cf*, where all the errors decrease. For *Cf*, the method based on metric learning for kernel regression is also validated to be effective. Hence we can conclude that metric learning can improve the forecasting results for all the considered roads.

4 Conclusion and Future Work

In this paper, metric learning based kernel regression is applied to short-term traffic flow forecasting. Comparisons with traditional kernel regression with the Euclidean metric under two criteria show that metric learning based kernel regression is effective for short-term traffic flow forecasting. We also propose a method to evaluate the effectiveness of learned metrics by analyzing eigenvalues of the corresponding metric matrices.

Though metric learning based kernel regression can improve prediction results, in order to know its full potential it is necessary to compare this method under the same conditions with other traffic flow forecasting methods, such as neural networks and the Bayesian network approach [8]. This will be investigated in the future.

Acknowledgments. This work was supported in part by the National Natural Science Foundation of China under Project 60703005, and in part by Shanghai Educational Development Foundation under Project 2007CG30.

References

1. William, B.M.: Modeling and Forecasting Vehicular Traffic Flow as a Seasonal Stochastic Time Series Process. Ph.D. Dissertation, Univ. Virginia, Charlottesville, VA (1999)
2. Moorthy, C.K., Ratcliffe, B.G.: Short Term Traffic Forecasting Using Time Series Methods. *Transp. Plan. Technol.* 12, 45–56 (1988)
3. Lee, S., Fambro, D.B.: Application of Subsets Autoregressive Integrated Moving Average Model for Short-Term Freeway Traffic Volume Forecasting. *Transp. Res. Rec.* 1678, 179–188 (1999)
4. Hall, J., Mars, P.: The Limitations of Artificial Neural Networks for Traffic Prediction. In: *Proc. 3rd IEEE Symp. Computers and Communications*, Athens, Greece, pp. 8–12 (1998)
5. Okutani, I., Stephanedes, Y.J.: Dynamic Prediction of Traffic Volume through Kalman Filter Theory. *Transp. Res., Part B: Methodol.* 18, 1–11 (1984)
6. Yu, G., Hu, J., Zhang, C., Zhuang, L., Song, J.: Short-Term Traffic Flow Forecasting based on Markov Chain Model. In: *Proc. IEEE Intelligent Vehicles Symp.*, Columbus, OH, pp. 208–212 (2003)
7. Müller, K.R., Smola, A.J., Rätsch, G., Schölkopf, B., Kohlmorgen, J., Vapnik, V.: Predicting Time Series with Support Vector Machines. In: *Proc. Int. Conf. Artificial Neural Networks*, pp. 999–1004 (1997)
8. Sun, S., Zhang, C.: A Bayesian Network Approach to Traffic Flow Forecasting. *IEEE Trans. Intell. Transp. Syst.* 7, 124–132 (2006)
9. Lam, W.H.K., Xu, J.: Estimation of AADT from Short Period Counts in Hong Kong—A Comparison between Neural Network Method and Regression Analysis. *J. Adv. Transp.* 34, 249–268 (2000)
10. Smith, B.L., Williams, B.M., Oswald, R.K.: Comparison of Parametric and Non-parametric Models for Traffic Flow Forecasting. *Transp. Res., Part C: Emerg. Technol.* 10, 303–321 (2002)
11. Davis, G.A., Nihan, N.L.: Non-Parametric Regression and Short-Term Freeway Traffic Forecasting. *J. Transp. Eng.* 177, 178–188 (1991)
12. Weinberger, K.Q., Tesauro, G.: Metric Learning for Kernel Regression. In: *Proc. 11th Int. Conf. Artificial Intelligence and Statistics*, Omnipress, Puerto Rico, pp. 608–615 (2007)
13. Duda, R.O., Hart, P.E., Stork, D.G.: *Pattern Classification*. John Wiley & Sons, New York (2000)

Hybrid Weighted Distance Measures and Their Application to Pattern Recognition

Zeshui Xu

School of Economics and Management
Southeast University, Nanjing 210096, China
xu_zeshui@263.net

Abstract. Distance measures are an important means to find the difference of data. In this paper, we develop a type of hybrid weighted distance measures which are based on the weighted distance measures and the ordered weighted averaging operator, and also point out some of their special cases. Then, we apply the developed measures to pattern recognition.

1 Introduction

In real life, we usually need to process various data information by means of distance measures. As a result, many authors have focused their attention on information processing, and introduced a variety of distance measures over the last decades [1-5]. Most the existing distance measures are the weighted distance measures, including some well-known distance measures such as the weighted Hamming distance and the weighted Euclidean distance, etc., All the distance measures of this type take the importance of each difference value into consideration. Recently, motivated by the idea of the ordered weighted averaging operator [6], Xu and Chen [7] introduced a type of ordered weighted distance measures whose fundamental aspect is the reordering step. The ordered weighted distance measures emphasize the importance of ordered position of each difference value rather than the importance of each difference value itself. Therefore, weights represent different aspects in both the above two types of distance measures. In order to reflect the importance of both the difference value and its ordered position, it is necessary to develop some new distance measures. To do so, in this paper, we develop a type of hybrid weighted distance measures which have the advantages of distance measures of both the above two types. We also discuss some special cases of the developed distance measures, and finally apply their to pattern recognition.

2 Hybrid Weighted Distance Measures

Among the existing weighted distance measures, the weighted Hamming distance and the weighted Euclidean distance are the two most widely used ones, which can be described as follows.

Let $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_n)$ and $\beta = (\beta_1, \beta_2, \dots, \beta_n)$ be two collections of arguments, and $w = (w_1, w_2, \dots, w_n)$ be the weight vector of the absolute difference values $|\alpha_j - \beta_j|$ ($j = 1, 2, \dots, n$), where $w_j \geq 0$, $j = 1, 2, \dots, n$, and $\sum_{j=1}^n w_j = 1$, then

1) The weighted Hamming distance:

$$WHD(\alpha, \beta) = \sum_{j=1}^n w_j |\alpha_j - \beta_j| \quad (1)$$

2) The weighted Euclidean distance:

$$WED(\alpha, \beta) = \sqrt{\sum_{j=1}^n w_j (\alpha_j - \beta_j)^2} \quad (2)$$

The following form is a generalization of both the distance measures (1) and (2):

$$WD(\alpha, \beta) = \left(\sum_{j=1}^n w_j |\alpha_j - \beta_j|^\lambda \right)^{1/\lambda}, \quad \lambda > 0 \quad (3)$$

All the above distance measures (1)-(3) take the importance of each argument (absolute difference values $|\alpha_j - \beta_j|$) into consideration.

Recently, motivated by the idea of the ordered weighted averaging operator [6], Xu and Chen [7] introduced a type of ordered weighted distance (OWD) measures:

$$OWD(\alpha, \beta) = \left(\sum_{j=1}^n v_j |\alpha_{\sigma(j)} - \beta_{\sigma(j)}|^\lambda \right)^{1/\lambda}, \quad \lambda > 0 \quad (4)$$

where $\sigma(1), \sigma(2), \dots, \sigma(n)$ is any permutation of $(1, 2, \dots, n)$, such that

$$|\alpha_{\sigma(j-1)} - \beta_{\sigma(j-1)}| \geq |\alpha_{\sigma(j)} - \beta_{\sigma(j)}|, \quad j = 2, 3, \dots, n \quad (5)$$

and $v = (v_1, v_2, \dots, v_n)$ is the weighting vector associated with the OWD measures (i.e., the weighting vector of the ordered positions of the arguments $|\alpha_j - \beta_j|$ ($j = 1, 2, \dots, n$)), $v_j \geq 0$, $j = 1, 2, \dots, n$, and $\sum_{j=1}^n v_j = 1$.

It is clear that the OWD measures (4) emphasize the importance of ordered position of each argument $|\alpha_j - \beta_j|$ rather than the importance of the argument itself.

Consider that weights represent different aspects in both the distance measures (3) and (4), in order to reflect the importance of both the argument $|\alpha_j - \beta_j|$ and its ordered position, here we develop a new type of distance measures as below:

Definition 1. Let $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_n)$ and $\beta = (\beta_1, \beta_2, \dots, \beta_n)$ be two collections of arguments, then we call

$$HWD(\alpha, \beta) = \left(\sum_{j=1}^n v_j \Delta(\alpha_{\sigma(j)}, \beta_{\sigma(j)}) \right)^{1/\lambda}, \quad \lambda > 0 \quad (6)$$

a type of hybrid weighted distance (HWD) measures, where $\Delta(\alpha_{\sigma(j)}, \beta_{\sigma(j)})$ is the j th largest of the weighted arguments $\Delta(\alpha_j, \beta_j)$ (here, $\Delta(\alpha_j, \beta_j) = nw_j |\alpha_j - \beta_j|^\lambda$, $j = 1, 2, \dots, n$), $v = (v_1, v_2, \dots, v_n)$ is the weighting vector associated with the HWD measures, $v_j \geq 0$, $j = 1, 2, \dots, n$, $\sum_{j=1}^n v_j = 1$, $w = (w_1, w_2, \dots, w_n)$ is the weight vector of the arguments $|\alpha_j - \beta_j|$ ($j = 1, 2, \dots, n$), $w_j \geq 0$, $\sum_{j=1}^n w_j = 1$, and n is the balancing coefficient which plays a role of balance (in such a case, if the vector (w_1, w_2, \dots, w_n) approaches $(1/n, 1/n, \dots, 1/n)$, then the vector $(nw_1 |\alpha_1 - \beta_1|^\lambda, nw_2 |\alpha_2 - \beta_2|^\lambda, \dots, nw_n |\alpha_n - \beta_n|^\lambda)$ approaches $(|\alpha_1 - \beta_1|^\lambda, |\alpha_2 - \beta_2|^\lambda, \dots, |\alpha_n - \beta_n|^\lambda)$).

Obviously, the HWD measures are also an extension of the traditional hybrid weighted aggregation operator [8], and the weighting vector $v = (v_1, v_2, \dots, v_n)$ can be determined by using some weight determining methods like the normal distribution based method, see [9] for more details.

Moreover, in what follows, we discuss two special cases of the HWD measures:

Theorem 1. If $v = (1/n, 1/n, \dots, 1/n)$, then the HWD measures (6) is reduced to the weighted distance (WD) measures (3).

Proof. Since $v = (1/n, 1/n, \dots, 1/n)$, then

$$HWD(\alpha, \beta) = \left(\sum_{j=1}^n v_j \Delta(\alpha_{\sigma(j)}, \beta_{\sigma(j)}) \right)^{1/\lambda} = \left(\frac{1}{n} \sum_{j=1}^n \Delta(\alpha_{\sigma(j)}, \beta_{\sigma(j)}) \right)^{1/\lambda}$$

$$\begin{aligned}
 &= \left(\frac{1}{n} \sum_{j=1}^n \Delta(\alpha_j, \beta_j) \right)^{1/\lambda} = \left(\frac{1}{n} \sum_{j=1}^n n w_j |\alpha_j - \beta_j|^\lambda \right)^{1/\lambda} \\
 &= \left(\sum_{j=1}^n w_j |\alpha_j - \beta_j|^\lambda \right)^{1/\lambda} \\
 &= WD(\alpha, \beta)
 \end{aligned} \tag{7}$$

which completes the proof of Theorem 1.

Theorem 2. If $w = (1/n, 1/n, \dots, 1/n)$, then the HWD measures (6) is reduced to the OWD measures (4).

Proof. Since $w = (1/n, 1/n, \dots, 1/n)$, then

$$\Delta(\alpha_j, \beta_j) = n w_j |\alpha_j - \beta_j|^\lambda = |\alpha_j - \beta_j|^\lambda, \quad j = 1, 2, \dots, n \tag{8}$$

thus

$$HWD(\alpha, \beta) = \left(\sum_{j=1}^n v_j \Delta(\alpha_{\sigma(j)}, \beta_{\sigma(j)}) \right)^{1/\lambda} = OWD(\alpha, \beta) \tag{9}$$

where $\Delta(\alpha_{\sigma(j)}, \beta_{\sigma(j)})$ is the j th largest of the weighted arguments $\Delta(\alpha_j, \beta_j)$ $j = 1, 2, \dots, n$. This completes the proof of Theorem 2.

From Theorems 1 and 2, it follows that the HWD measures generalize both the WD measures (3) and the OWD measures (4), and thus, they can reflect the importance of both the considered argument and its ordered position.

In the next section, we shall apply the HWD measure (6) to pattern recognition.

3 Application of the HWD Measures to Pattern Recognition

Assume that there exist m patterns, which are represented by $\alpha_i = (\alpha_i(x_1), \alpha_i(x_2), \dots, \alpha_i(x_n))$ ($i = 1, 2, \dots, m$) in the feature space $X = \{x_1, x_2, \dots, x_n\}$, and $w = (w_1, w_2, \dots, w_n)$ is the weight vector of x_j ($j = 1, 2, \dots, n$), where $w_j \geq 0$, $j = 1, 2, \dots, n$, and $\sum_{j=1}^n w_j = 1$. Moreover, suppose that there is a sample $\beta = (\beta(x_1), \beta(x_2), \dots, \beta(x_n))$ to be recognized in the feature space X .

We can utilize the HWD measures (6) to calculate the distance $HWD(\alpha_j, \beta)$ between each pattern α_j and the sample β , and then let

$$HWD(\alpha_{j_0}, \beta) = \min_{1 \leq j \leq m} \{HWD(\alpha_j, \beta)\} \quad (10)$$

hence, by (10), we can determine that the sample β belongs to the pattern α_{j_0} according to the principle of the minimum distance.

Below we employ a numerical example to illustrate the application of the above procedure.

Example. Assume that there are four patterns, which are represented by $\alpha_j (j=1, 2, 3, 4)$ in the feature space $X = \{x_1, x_2, \dots, x_7\}$:

$$\begin{aligned} \alpha_1 &= (80, 50, 60, 70, 100, 95, 40), \quad \alpha_2 = (100, 95, 65, 80, 90, 90, 50) \\ \alpha_3 &= (50, 90, 70, 90, 60, 100, 70), \quad \alpha_4 = (90, 70, 95, 75, 80, 65, 60) \end{aligned}$$

and the weight vector of the feature space $X = \{x_1, x_2, \dots, x_7\}$ is

$$w = (0.10, 0.15, 0.10, 0.20, 0.15, 0.20, 0.10)$$

Let

$$\beta = (70, 80, 90, 65, 90, 85, 95)$$

be a sample to be recognized. Then we utilize the OWD measures (6) (suppose that the weighting vector associated with the HWD measures is $v = (0.07, 0.13, 0.19, 0.22, 0.19, 0.13, 0.07)$, which is derived by the normal distribution based method [9]) to calculate the distance between α_j and β (without loss of generality, here we only take into consideration the cases of $\lambda = 1, 2$):

1) If $\lambda = 1$, then

$$HWD(\alpha_1, \beta) = 17.255, \quad HWD(\alpha_2, \beta) = 16.678$$

$$HWD(\alpha_3, \beta) = 19.600, \quad HWD(\alpha_4, \beta) = 14.490$$

thus

$$HWD(\alpha_4, \beta) = \min_{1 \leq j \leq 4} \{HWD(\alpha_j, \beta)\}$$

2) If $\lambda = 2$, then

$$HWD(\alpha_1, \beta) = 22.648, HWD(\alpha_2, \beta) = 20.223$$

$$HWD(\alpha_3, \beta) = 20.719, HWD(\alpha_4, \beta) = 15.863$$

thus,

$$HWD(\alpha_4, \beta) = \min_{1 \leq j \leq 4} \{HWD(\alpha_j, \beta)\}$$

The results of both 1) and 2) show that the sample β belongs to the pattern α_4 .

If we utilize the WD measures (3) and the OWD measures (4) to calculate the distance between each given pattern α_j and the sample β , then we can derive the following results, listed in Tables 1 and 2, respectively:

Table 1. The distances derived by the WD measures

	$WD(\alpha_1, \beta)$	$WD(\alpha_2, \beta)$	$WD(\alpha_3, \beta)$	$WD(\alpha_4, \beta)$
$\lambda = 1$	18.50	16.25	20.50	15.00
$\lambda = 2$	24.03	20.95	21.51	17.18

Table 2. The distances derived by the OWD measures

	$OWD(\alpha_1, \beta)$	$OWD(\alpha_2, \beta)$	$OWD(\alpha_3, \beta)$	$OWD(\alpha_4, \beta)$
$\lambda = 1$	19.20	18.60	20.95	14.60
$\lambda = 2$	23.57	21.75	21.52	16.37

From Tables 1 and 2, it is clear that all the results derived by the WD measures (3) and the OWD measures (4) show that the sample β also belongs to the pattern α_4 .

Among the WD, OWD and WHD measures, the WHD measures can not only reflect the importance of each argument, but also consider the importance of the ordered position of the argument, and thus remain the most original information in the final decision results.

4 Conclusions

The weighted distance (WD) measures and the ordered weighted distance (OWD) measures are the main two types of distance measures in the existing literature. By combining the advantages of these two types of distance measures, we have developed a new type of distance measures called hybrid weighted distance (HWD) measures, which can reflect the importance of both the considered argument and its ordered position. Both the WD and OWD measures are the special cases of the HWD

measures. We have also given an application of the HWD measures to pattern recognition. In future research, applying the HWD measures to other fields such as decision making, medical diagnosis, data mining, etc., may be interesting.

Acknowledgement

The work was supported by the National Science Fund for Distinguished Young Scholars of China (No.70625005), and the National Natural Science Foundation of China (No.70571087).

References

1. Bogart, K.P.: Preference structures II: Distances between asymmetric relations. *SIAM Journal on Applied Mathematics* 29, 254–262 (1975)
2. Nadler Jr., S.B.: *Hyperspaces of Sets*. Marcel Dekker, New York (1978)
3. Zwick, R., Carlstein, E., Budescu, D.V.: Measures of similarity among fuzzy concepts: a comparative analysis. *International Journal of Approximate Reasoning* 1, 221–242 (1987)
4. Kacprzyk, J.: *Multistage Fuzzy Control*. Wiley, Chichester (1997)
5. Xu, Z.S., Chen, J.: An overview of distance and similarity measures of intuitionistic fuzzy sets. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 16, 529–555 (2008)
6. Yager, R.R.: On ordered weighted averaging aggregation operators in multi-criteria decision making. *IEEE Transactions on Systems, Man and Cybernetics* 18, 183–190 (1988)
7. Xu, Z.S., Chen, J.: Ordered weighted distance measure. *Journal of Systems Science and Systems Engineering* 17 (in press, 2008)
8. Xu, Z.S., Da, Q.L.: An overview of operators for aggregating information. *International Journal of Intelligent Systems* 18, 953–969 (2003)
9. Xu, Z.S.: An overview of methods for determining OWA weights. *International Journal of Intelligent Systems* 20, 843–865 (2005)

A Multitask Learning Approach to Face Recognition Based on Neural Networks

Feng Jin and Shiliang Sun

Department of Computer Science and Technology,
East China Normal University, Shanghai 200241, P.R. China
fjin07@gmail.com, s1sun@cs.ecnu.edu.cn

Abstract. For traditional human face based biometrics, usually one task (face recognition) is learned at one time. This single task learning (STL) approach may neglect potential rich information resources hidden in other related tasks, while multitask learning (MTL) can make full use of the latent information. MTL is an inductive transfer method which improves generalization by using the domain information contained in the training signals of related tasks as an inductive bias. In this paper, backpropagation (BP) network based MTL approach is proposed for face recognition. The feasibility of this approach is demonstrated through two different face recognition experiments, which show that MTL based on BP neural networks is more effective than the traditional STL approach, and that MTL is also a practical approach for face recognition.

Keywords: multitask learning (MTL), single task learning (STL), face recognition, backpropagation (BP), artificial neural network (ANN).

1 Introduction

Face recognition research has many theoretical and practical significances, for example, it can facilitate the development of some related disciplines, and provide a practical means for biometrics [1]. Compared to other biometric technologies, such as fingerprint and iris recognition, the availability of face recognition technology has its unique advantages, for example, the capture of face images usually does not intervene people's normal activities. It is because of this that over the last decade, face recognition has become a popular research area in computer vision and one of the most successful applications of image analysis and understanding. Up to the present, people have introduced a variety of methods for face recognition, such as those based on principal component analysis (PCA), independent component analysis (ICA) and linear discriminant analysis (LDA) [2],[3],[4]. In this paper, exploring the performance of face recognition using neural networks from a new point of view is our concern.

Artificial neural network (ANN) is a mathematical model to deal with information processing using a structure similar to the connection of brain nerves [5]. Theoretically, neural networks (NN) can fully approximate arbitrary complex linear or nonlinear relations, with adopting a parallel distributed processing

method which makes rapid and mass calculation possible. For image recognition, a NN can take many given images as its inputs and learns to give outputs which approximate the corresponding image categories as accurately as possible. The traditional neural network approach for face recognition is to learn a task at a time. It is a single task learning (STL) model which neglects potential rich information resources hidden in other related tasks.

In fact for many practical situations, a classification task can often relate to several other tasks. Since related tasks tend to share common factors, solving them together is expected to be more advantageous than solving them independently. This approach is called multi-task learning (MTL) and has been theoretically and experimentally proven to be useful [6],[7]. In MTL, the task considered most is called the main task, while others are called extra tasks. MTL can improve generalization performance of neural networks by utilizing some field-specific training information contained in the extra tasks [8]. Of course, MTL can be applied to different kinds of learning machines, such as support vector machine (SVM), decision trees, and so on. However, to the best of our knowledge, there are almost no results reported on face recognition using multiple learning tasks, for example, training a classifier to identify faces and poses simultaneously. Here, we only consider neural networks, though a similar approach can be extended to other learning machines. In this paper, MTL backpropagation (BP) networks are adopted to carry out face recognition. Experiments with encouraging results show that this approach is considerably effective for the considered face recognition application.

The rest of this paper is organized as follows. MTL NN and its benefits for face recognition are introduced in Section 2. Then we give the model construction mechanism, and report experimental results in Section 3. Section 4 summarizes this paper and gives future research directions.

2 MTL NN for Face Recognition

Normally, most learning methods such as traditional neural networks only have one task. This is because when solving a complicated problem, we usually split it into a number of small, appropriately independent subproblems to learn [9]. In fact, this ideology ignores the probably close relationship among different tasks, and therefore is not optimal.

MTL is a form of inductive transfer whose main goal is to improve generalization performance using the domain-specific information included in the training signals of extra tasks. When MTL is used for face recognition, important improvements at two different levels can be achieved. One is that the number of training samples needed to learn each classification tasks decreases as more related tasks are learned [10]. The other is that it has been proved that under some theoretic conditions, a classifier trained on sufficiently related tasks is likely to find good solutions to solve novel related tasks [10]. So in the present study, we consider the MTL paradigm which can simultaneously learn related tasks together. In MTL NN, the main task is the one considered most, and the extra tasks trained at the same time only serve as an inductive bias. Of course, extra

tasks should be chosen to have a certain relationship with main task. In fact, the training signals of the extra tasks serve as an inductive bias which is used to improve the generalization accuracy in order to well complete the main task. In this paper, classifying faces images is selected as the main task and distinguishing the directions of faces is chosen as the extra task in MTL NN. Both tasks are trained in parallel using a shared representation. And it is expected that the information contained in these extra training signals can help the hidden layer learn a better internal representation for the main task.

3 Model Construction and Experiments

In this section, we show the usefulness of the proposed method through two experiments. The first one is implemented in the ideal circumstance where all types of postures (face directions) in the test set are completely included in the training set, illuminating that whether MTL is superior to STL in terms of performance. The second experiment is to illustrate the practicality: when postures in the test set appear brand new, whether MTL can improve the performance.

3.1 Experiment One

Data Description. The data used for analysis are from a face database which has images of 20 different people, including approximately 32 images per person, varying the directions in which they are looking (left, right, straight ahead, up) [11]. In total, 624 greyscale images are collected, each with a resolution of 120×128 . Generally for algorithm evaluation, samples are divided into three disjoint parts: the training set, the validation set and the test set. The training set is used to estimate models. The validation set is used to determine the network structure and the parameters which control the complexity degree of the model. And the test set is to validate the performance of model selected ultimately. A typical partition is that the training set occupies 50 per cent of the total samples, while the validation set and the test set take up 25 per cent respectively. In this experiment, 16 images including four different directions of each person are chosen as training set, totaling 320 images. Then we select 8 images which also include four different directions from each person as the verification set. The rest are used for test data. It means that a neural network is trained using the data set which contains all face directions of each person. The purposes of this experiment are making classification based on the same faces images using MTL and STL, and comparing accuracy rates of face recognition in these different methods.

Model Design. A three-layer NN is chosen. The reason is that it has ability to approximate to any continuous function, as long as the appropriate number of hidden layer neurons and right activation functions are used. The model building can be concluded as follows:

(1). The design option chosen in this case is instead to encode the image as a fixed set of 30×32 pixel intensity values. The 30×32 pixel image is, in fact, a coarse resolution summary of the original 120×128 captured image, with each coarse pixel intensity calculated as the mean of the corresponding high-resolution pixel intensities [11]. Each pixel corresponds to one network input, totaling 960 inputs. In order to accelerate the learning speed of NN, the input signal can be normalized.

(2). Sigmoid function is selected as the specific activation function between input layer and hidden layer. The form of sigmoid function is described as

$$f(x) = \frac{1}{1 + e^{-x}} . \quad (1)$$

This function can make neuronal inputs map to a range from 0 to 1. It is suitable to train NN using the algorithm of BP, since $f(x)$ is a differentiable function. The activation function between hidden layer and output layer is a linear function whose form is shown as

$$f(x) = x . \quad (2)$$

It is also a differentiable function which can get the same range as its inputs. A network training function *traindxx* is selected which can update weight and bias values according to gradient descent momenta and adaptive learning rates.

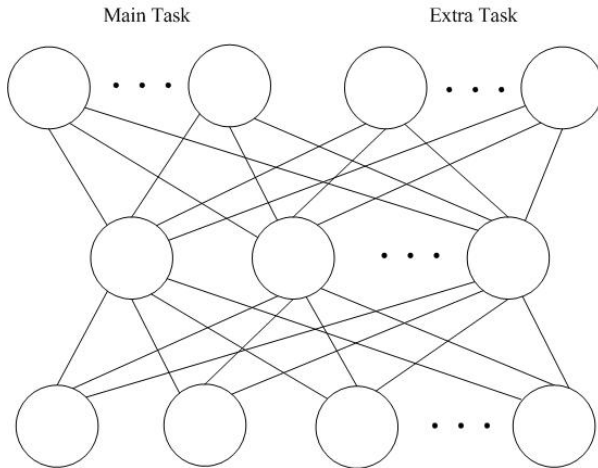


Fig. 1. NN using MTL for face recognition

(3). The MTL NN adopts the 1-of-n output encoding and has 24 output units, including main task which identifies the species of input images and extra task which learns the direction in which the person is facing. Twenty distinct output units are used as main task learning, each representing one of the twenty possible face categories, with the highest-valued output taken as the network prediction.

The encoding of extra task is the same as the main task, the difference between them is that four output units are only needed for extra task learning. The constructed MTL network model is given in Fig. 1. STL is employed as a comparative method and the STL network model is given in Fig. 2. The net only has one task using 20 output units. The difference between those two NNs exists only in their output layers. For the hidden layer, 12 neurons in MTL NN are selected to respectively compare with STL NNs which have 10, 12 and 14 neurons.

(4). There are many parameters in the training function including the largest training epoch, training time, and network error target which can be all acted as the stopping conditions of training. Choosing correctly parameters can be advantageous to establish a better neural network and is much easier to achieve the expected performance. Here, the largest training epoch is chosen as the cessation condition of the neural network based on observing the mean squared error (MSE) of the validation set. Furthermore, the validation set is used to prevent the possible overfitting phenomenon in NN learning. The parameter of learning rate decides the change of value generated in every cycle of training. Big learning rate may induce system instability, but a small learning rate will lead to a longer learning time and slower convergence rate. In this learning experiment, the initial learning rate is set to 0.3.

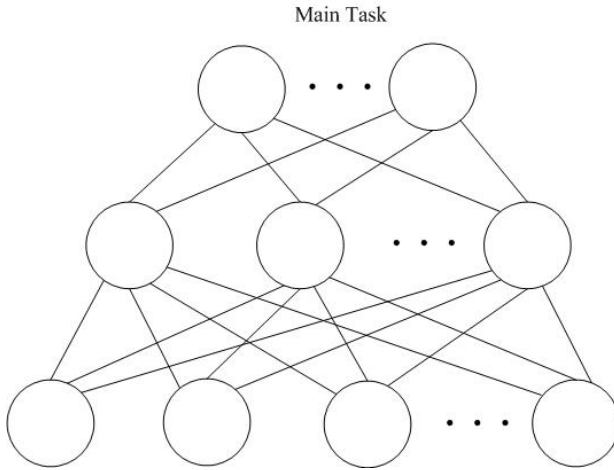


Fig. 2. NN using STL for face recognition

Result. Each experiment runs five times, and the median is selected as the final result. The performance is evaluated in the test set which has 144 face images. The results of this experiment are exhibited in Table 1, showing that the number of face images classified correctly has considerably risen after MTL is used. Though this test is biased in favor of STL because it compares single runs of MTL on an unoptimized net size with several independent runs of STL that use different hidden units and are able to find near-optimal net size. MTL can successfully capture gist which is needed for precise face recognition by making

Table 1. Performance of STL and MTL with different hidden layers on tasks of face recognition in experiment one

	STL			MTL
Number of Hidden Unit	10	12	14	12
Number of Correct Classifications	96	97	117	133
The Accuracy Rate	67%	67%	81%	91%

use of the information from other tasks. It implies that more information is provided by extra task for the main task learning and the identifying precision is improved. Therefore, MTL NN can be used for face recognition to get relatively precise results. However, the experiment is based on both the training set and the test set which have all face directions. Thus, another experiment which is changed in the training set is given below to view whether the proposed method can also effectively exploit the information of related tasks.

3.2 Experiment Two

Data used in this experiment are from the same face database and the different points compared to the experiment one are the choices of training set, certification set and test set. Only two kinds of people face directions (left, straight ahead) are selected from each category as training set in this experiment, totaling 320 images. From the remaining, 160 pictures are selected as verification set, and the rest are used as test set. It means that only two face gestures (left, straight ahead) images data are used to design neural network. Then we observe the classification performance whether better than single task NN in the same training data when the samples of new postures are added. The main task of MTL NN is also to identify the species of input images using 20 output units. And two output units are used for extra task, because only images of two face directions (left, straight ahead) are selected as inputs during the course of designing the NN. STL is also employed as a comparative method. STL NN has 20 output units for learning one task which is to identify the species of input images.

Result. In the experiment, the number of hidden layer’s neurons is increased in order to gain better precision. The MTL net has 18 hidden units, and the STL nets have 15, 18, or 21 hidden units. This experiment also runs five times, the median is selected as the final result. The performance is evaluated in the test set which has 151 face images.

From the information given in Table 2, we can include that when MTL and STL NN have the same hidden units, performance of the former is better than the latter. Besides, the performance of STL NN can not catch up with MTL NN, despite the

Table 2. Performance of STL and MTL with different hidden layers on tasks of face recognition in experiment two

	STL	MTL
Number of Hidden Unit	15 18 21	18
Number of Correct Classifications	103 128 134	135
The Accuracy Rate	68% 84% 88%	89%

number of hidden units in STL NN increases to 21. It is under the condition that training a NN using several face direction images, but testing the NN using else face direction images. It implies that generalization performance is improved when using MTL. The ability of practical application is to be fully exhibited.

4 Conclusion

In this paper, we propose the approach of multi-task learning to face recognition, which overcomes the limitation of existing approaches such as STL by making full use of information contained in the extra tasks. We demonstrated through experiments that the proposed method is useful in face recognition; moreover, it also works well when new face postures appear during testing. Experimental results have demonstrated the superiority offered by MTL. The superior performance of MTL on face image recognition shows that manifold information from related tasks can play positive and complementary effects in real applications, suggesting that one can find significant benefits in practice by performing MTL.

For different face databases, different tasks can be selected as extra tasks whose roles are to serve as inductive biases for the main task. In the future, developing novel algorithms for face recognition using multitask learning by means of k-nearest neighbor and SVM can be investigated..

Acknowledgments. This work was supported in part by the National Natural Science Foundation of China under Project 60703005, and in part by Shanghai Educational Development Foundation under Project 2007CG30.

References

1. Zhao, W., Chellappa, R., Phillips, P.J., Rosenfeld, A.: Face Recognition: A Literature Survey. *ACM Computing Surveys* 35(4), 399–458 (2003)
2. Bellhumeur, P., Hespanha, J., Kriegman, D.: Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19(7), 771–720 (1997)

3. Turk, M., Pentland, A.: Eigenfaces for Recognition. *Journal of Cognitive Neuroscience* 3(1), 71–86 (1991)
4. Turk, M., Pentland, A.: Face Recognition using Eigenfaces. In: *Proc. of the IEEE Conf. On Computer Vision and Pattern Recognition*, pp. 586–591 (1991)
5. Haykin, S.: *Neural Networks: A Comprehensive Foundation*, 1st edn. Prentice Hall PTR, Englewood Cliffs (1994)
6. Evgeniou, T., Pontil, M.: Regularized multitask learning. In: *Proc. of 17th SIGKDD Conf. on Knowledge Discovery and Data Mining* (2004)
7. Micchelli, C.A., Pontil, M.: Kernels for multi-task learning. *Advances in Neural Information Processing Systems* 17, 921–928 (2005)
8. Lindbeck, A., Snower, D.J.: Multi-task Learning and the Reorganization of Work. *Journal of Labor Economics* 18(3), 353–376 (2000)
9. Caruana, R.: Multitask learning. In: *Proceedings of the 10th International Conference on Machine Learning, ML 1993*, University of Massachusetts, Amherst, pp. 41–48 (1993)
10. Baxter, J.: A model of inductive bias learning. *Proc. Journal of Machine Learning Research* 12, 149–198 (2000)
11. Mitchell, T.: *Machine Learning*, 2nd edn. McGraw Hill, New York (1997)

Logic Synthesis for FSMs Using Quantum Inspired Evolution

Marcos Paulo Mello Araujo¹, Nadia Nedjah¹, and Luiza de Macedo Mourelle²

¹ Department of Electronics Engineering and Telecommunication

² Department of Systems Engineering and Computation,
Engineering Faculty, State University of Rio de Janeiro, Brazil

Abstract. Synchronous finite state machines are very important for digital sequential systems. Among other important aspects, they represent a powerful way for synchronising hardware components so that these components may cooperate adequately in the fulfilment of the main objective. In this paper, we propose to use an evolutionary methodology inspired from quantum computation to yield a concise and efficient evolvable hardware that implements the state machine control logic.

1 Introduction

Traditionally, the design process of a state machine goes through five main steps: *(i)* the specification of the sequential system, which should determine the next states and outputs of every present state of the machine. This is done using state tables and state diagrams; *(ii)* the state reduction, which should reduce the number of present states using equivalence and output class grouping; *(iii)* the state assignment, which should assign a distinct combination to every present state. *(iv)* the minimisation of the control combinational logic using K-maps and transition maps; *(v)* finally, the implementation of the state machine, using gates and flip-flops.

In this paper, we concentrate on the fourth step of the design process, i.e. the control logic synthesis and minimisation. We present a quantum inspired evolutionary algorithm designed to evolve the circuit that controls the machine current and next states and to provide its outputs.

The remainder of this paper is organised into four sections. In Section 2, we briefly introduce the principles of quantum computation and present a quantum inspired synthesiser for evolving optimal control logic circuit provided the state assignment for the specification of the state machine in question. After that, in Section 3, we describe the circuit encoding, quantum gates used as well as the fitness function, which determines whether a control logic design is better than another and how much. Then, in Section 4, we compare the area and time requirements of the designs evolved through our evolutionary synthesiser for some well-known benchmarks and compare the obtained results with those obtained using the traditional method to design state machine.

2 Quantum Inspired Evolutionary Algorithm

As any evolutionary algorithms, this algorithm is based on a population of solutions which is maintained through many generations. It seeks the best fitted solution to the problem, evaluating the characteristics of those included in the current population. In the next sections, we describe the quantum inspired representation of the individual and the underlying computational process.

2.1 Principles of Quantum Computing

In quantum computing, the smallest unit of information stored in a two-state system is called a quantum bit or qubit [5]. The 0 and 1 states of a classical bit, are replaced by the state vectors $|0\rangle$ and $|1\rangle$ of a qubit. These vectors are usually written using the *bracket* notation, introduced by Paul Dirac. The state vectors of a qubit are represented as in (1)

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad \text{and} \quad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \quad (1)$$

While the classical bit can be in only one of the two basic states that are mutually exclusive, the generic state of one qubit can be represented by the linear combination of the state vectors $|0\rangle$ and $|1\rangle$, as $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, wherein α and β are complex numbers. The state vectors $|0\rangle$ and $|1\rangle$ form a canonical base and the vector $|\psi\rangle$ represents the superposition of these vectors, with α and β amplitudes. The unit normalization of the state of the qubit ensures that $|\alpha|^2 + |\beta|^2 = 1$ is true. The phase of a qubit is defined by an angle ζ as $\zeta = \arctan(\beta/\alpha)$ and the product $\alpha \cdot \beta$ is represented by the symbol d and defined as $d = \alpha \cdot \beta$, where d stands for the quadrant of qubit phase ζ . If d is positive, the phase ζ lies in the first or third quadrant; otherwise, the phase ζ lies in the second or fourth quadrant [9].

The physical interpretation of the qubit is that it may be simultaneously in the states $|0\rangle$ and $|1\rangle$, which allows that an infinite amount of information could be stored in state $|\psi\rangle$. However, in the act of observing a quantum state, it collapses to a single state [7]. The qubit collapses to state 0, with probability $|\alpha|^2$ or state 1, with probability $|\beta|^2$. A system with m qubits contains information on 2^m states. The linear superposition of possible states is shown in (2)

$$|\psi\rangle = \sum_{k=1}^{2^m} C_k |S_k\rangle, \quad (2)$$

wherein C_k specifies the probability amplitude of the corresponding states S_k and subjects to the normalization condition of $|C_1|^2 + |C_2|^2 + \dots + |C_{2^m}|^2 = 1$.

The state of a qubit can be changed by the operation of a quantum gate or Q-gate. The Q-gates apply a unitary operation U on a qubit in the state $|\psi\rangle$ making it evolve to the state $U|\psi\rangle$, which maintains the probabilities interpretation as defined before. There are several Q-gates, such as the *NOT* gate, *Controlled-NOT* gate, *Hadamard* gate, *rotation* gate, etc.

2.2 Quantum Inspired Representation

The evolutionary algorithms, like the genetic algorithms, for instance, can use several representation that have been used with success: binary, numeric and symbolic representation [6]. The quantum inspired evolutionary algorithms use a new representation, that is a probabilistic representation based on the concept of qubits and a q-individual as a string of qubits. A q-individual can be defined as in [3] wherein $|\alpha_i|^2 + |\beta_i|^2 = 1$, for $i = 1, 2, 3, \dots, m$.

$$p = \left[\begin{array}{c|c|c|c} \alpha_1 & \alpha_2 & \alpha_3 & \dots & \alpha_m \\ \beta_1 & \beta_2 & \beta_3 & \dots & \beta_m \end{array} \right] \quad (3)$$

The advantage of the representation of the individuals using qubits instead of the classical representation is the capacity of representing the linear superposition of all possible states. The evolutionary algorithms with the quantum inspired representation of the individual should present a population diversity better than other representations, since they can represent the linear superposition of states [2] [4]. Only one q-individual of n q-bits is enough to represent 2^n states. Using the classical representation, 2^n individuals would be necessary.

2.3 Algorithm Description

The basic structure of the quantum inspired evolutionary algorithm presented in this paper is described by Algorithm 1. The quantum inspired evolutionary algorithms maintain a population of q-individuals, $P(g) = \{p_1^g, p_2^g, \dots, p_n^g\}$ at generation g , where n is the size of population, and p_j^g is a q-individual defined as in [4], where m is the number of qubits, which defines the string length of the q-individual, and $j = 1, 2, \dots, n$.

Algorithm 1. Quantum Inspired Evolutionary Algorithm

1. $g := 0$; **generate** initial population P_0 with n individuals;
 2. **observe** P_0 into S_0 ;
 3. **evaluate** the fitness of every solution in S_0 ; **store** S_0 into B_0 ;
 4. **while** (**not** *termination condition*) **do**
 5. $g := g + 1$; **observe** P_{g-1} into S_g ;
 6. **evaluate** the fitness of every solution in S_g ; **update** P_g using a Q-gate;
 7. **apply** probability constraints; **store** best solutions among B_{g-1} and S_g into B_g ;
 8. **store** the best solution in B_g into b ;
 9. **if** (*no improvement for many generations*) **then**
 10. **replace** all the solution of B_g by b ;
 11. **end if**
 12. **end while**
-

$$p_j^g = \left[\begin{array}{c|c|c|c} \alpha_{j_1}^g & \alpha_{j_2}^g & \alpha_{j_3}^g & \dots & \alpha_{j_m}^g \\ \beta_{j_1}^g & \beta_{j_2}^g & \beta_{j_3}^g & \dots & \beta_{j_m}^g \end{array} \right], \quad (4)$$

The initial population of n individuals is generated setting $\alpha_i^0 = \beta_i^0 = 1/\sqrt{2}$ ($i = 1, 2, \dots, m$) of all $\mathbf{p}_j^0 = \mathbf{p}_j^g|_{g=0}$ ($j = 1, 2, \dots, n$). This allows each q-individual to be the superposition of all possible states with the same probability.

The binary solutions in S_g are obtained by an observation process of the states of every q-individual in P_g . Let $S_g = \{\mathbf{s}_1^g, \mathbf{s}_2^g, \dots, \mathbf{s}_n^g\}$ at generation g . Each solution, \mathbf{s}_i^g for ($i = 1, 2, \dots, n$), is a binary string with the length m , that is, $\mathbf{s}_i^g = s_1 s_2 \dots s_m$, where s_j for ($j = 1, 2, \dots, m$) is either 0 or 1.

The observation process is implemented using random probability: for each pair of amplitudes $[\alpha_k, \beta_k]^T$ ($k = 1, 2, \dots, n \times m$) of every qubit in the population P_g , a random number r in the range $[0, 1]$ is generated. If $r < |\beta_k|^2$, the observed qubit is 1; otherwise, it is 0.

The q-individuals in P_g are updated using a Q-gate, which is detailed in later. We impose some probability constraints such that the variation operation performed by the Q-gate avoid the premature convergence of a qubits to either to 0 or 1. This is done by not allowing neither of $|\alpha|^2$ nor $|\beta|^2$ to reach 0 or 1. For this purpose, the probability $|\alpha|^2$ and $|\beta|^2$ are constrained to 0.02 as a minimum and 0.98 as a maximum. Such constraints allows us to escape local minima.

After a given number of generations, if the best solution b does not improve, all the solutions stored into B_g are replaced by b . This step can induce a variation of the probabilities of the q-individuals. This also allows the algorithm to escape local minima and avoid the stagnant state.

3 Application in Evolvable Logic Synthesis

Exploiting quantum inspired evolutionary algorithm, we automatically generate novel control logic circuits that are reduced with respect to area and time requirements. The allowed gates are NOT, AND, OR, XOR, NAND, NOR, XNOR and WIRE. The latter represents a physical wire and thus, the absence of a gate.

3.1 Circuit Encoding

We encode circuit designs using a matrix of cells that may be interconnected. A cell may or may not be involved in the circuit schematics and consists of two inputs, a logical gate and a single output. A cell draws its input signals from the outputs of previous column. The cells located in the first column draw their inputs from the circuit global input signals. Each cell is encoded with a number of qubits, enough to represent the allowed gates and the signals that may be connected in each input of the cell. Note that the total number of qubits may vary depending on the number of outputs of the previous column or the number of primary inputs in the case of the first column. An example of a matrix of cells with respect to this encoding is given in Fig. 1. Fig. 2-(a) represents a cell encoding and a possible observation of the qubits states and the Fig. 2-(b) indicates the correspondent circuit encoded by this cell, that is composed by an AND gate with its input A and B connected to the first and third element of its previous column.

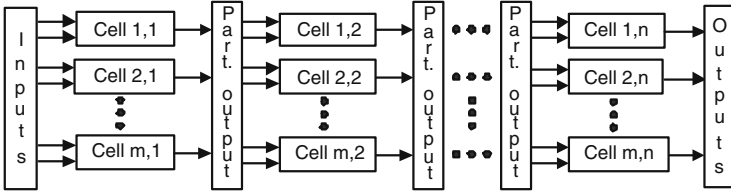


Fig. 1. Encoded circuit schematics

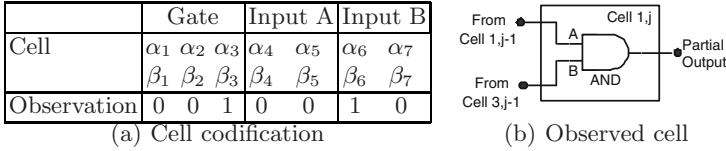


Fig. 2. Example of a cell with 4 output signals in previous column

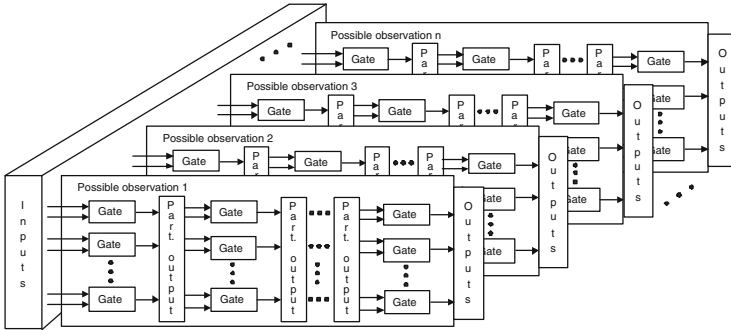


Fig. 3. Encoded circuit schematics

When the observation of the qubits that define the gate yields the code of WIRE, then the signal connected to the cells *A* input appears in the partial output of the cell. When the number of partial outputs of a column or the global inputs are not a power of 2, some of them are repeated in order to avoid that a cell be mapped to an inexistent input signal. The circuit primary output signals are the output signals of the cells in the last column of the matrix. If the number of global outputs are less than the number of cells in the last column, then some of the output signal are not used in the evolutionary process.

The power of the quantum inspired representation can be evidenced by the Fig. 3, which shows that all possible circuits can be represented with only one *q*-individual in a probabilistic way, as explained in the Section 2.2.

The number of *q*-individual included (population size) as well as the number of cells per *q*-individual are parameters that should be adjusted considering

the state machine complexity. The complexity depends on the number of inputs, outputs, states and states transitions of the machine.

3.2 Q-gate for Evolvable Hardware

To drive the individuals toward better solutions, a Q-gate is used as a variation operator of the quantum inspired evolutionary algorithm presented at this paper. After an update operation, the qubit must always satisfy the normalization condition $|\alpha'|^2 + |\beta'|^2 = 1$, where α' and β' are the amplitudes of the new qubit.

Initially, each q-individual represents all possible states with the same probability. As the probability of every qubit approaches either 1 or 0 by the Q-gate, the q-individual converges to a single state and the diversity property disappears gradually. By this mechanism, the quantum inspired evolutionary algorithm can treat the balance between exploration and exploitation [4]. The Q-gate used is inspired by a quantum rotation gate. This is defined in (5)

$$\begin{bmatrix} \alpha' \\ \beta' \end{bmatrix} = \begin{bmatrix} \cos(\Delta\theta) & -\sin(\Delta\theta) \\ \sin(\Delta\theta) & \cos(\Delta\theta) \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix}, \quad (5)$$

where $\Delta\theta$ is the rotation angle of each qubit toward states 0 or 1 depending on the amplitude signs. The angle $\Delta\theta$ should be adjusted to application problem.

The value of the angle $\Delta\theta$ can be selected from the Table 1, where $f(\mathbf{s}_i^g)$ and $f(\mathbf{b}_i^g)$ are the fitness values of \mathbf{s}_i^g and \mathbf{b}_i^g , and s_j and b_j are the j th bits of the observed solutions \mathbf{s}_i^g and the best solutions \mathbf{b}_i^g , respectively.

The rotation gate allows to change the amplitudes of the considered qubit, as follows: (i) If s_j and b_j are 0 and 1, respectively, and if $f(\mathbf{s}_i^g) \geq f(\mathbf{b}_i^g)$ is false then if the qubit is located in the first or third quadrant as defined before, θ_3 , the value of $\Delta\theta$ is set to a positive value to increase the probability of the state $|1\rangle$; Otherwise, if the qubit is located in the second or fourth quadrant, $-\theta_3$ should be used to increase the probability of the state $|1\rangle$; On the other hande, if s_j and b_j are 1 and 0, respectively, and if $f(\mathbf{s}_i^g) \geq f(\mathbf{b}_i^g)$ is false and the qubit is located in the first or third quadrant, θ_5 is set to a negative value to increase the probability of the state $|0\rangle$; Otherwise, i.e. if the qubit is located in the 2nd or 4thquadrant, $-\theta_5$ should be used to increase the probability of state $|0\rangle$.

When it is ambiguous to select a positive or a negative number for the angle parameters, we set it the values to zero as recommended in [4]. The magnitude of $\Delta\theta$ has an effect on the speed of convergence. If it is too big, the search grid of the algorithm would be large and the solutions may diverge or converge prematurely to a local optimum. If it is too small, the search grid of the algorithm would be small and the algorithm may fall in stagnant state. Hence, the magnitude of $\Delta\theta$ is defined as a variable, which values depend on the application problem.

3.3 Circuit Fitness Evaluation

To evaluate the fitness of each solution, some constraints were considered: First of all, the evolved specification must obey the input/output behaviour, which is given in a tabular form of expected results given the inputs. This is the truth

Table 1. Look-up table of $\Delta\theta$

s_j	b_j	$f(\mathbf{s}_i^g) \geq f(\mathbf{b}_i^g)$	$\Delta\theta$	s_j	b_j	$f(\mathbf{s}_i^g) \geq f(\mathbf{b}_i^g)$	$\Delta\theta$
0	0	false	θ_1	1	0	false	θ_5
0	0	true	θ_2	1	0	true	θ_6
0	1	false	θ_3	1	1	false	θ_7
0	1	true	θ_4	1	1	true	θ_8

table of the expected circuit. Secondly, the circuit must have a reduced size. This constraint allows us to yield compact digital circuits. Finally, the circuit must also reduce the signal propagation delay. This allows us to reduce the response time and so discover efficient circuits.

We estimate the necessary area for a given circuit using the concept of gate equivalent. This is the basic unit of measure for digital circuit complexity [3]. It is based upon the number of logic gates that should be interconnected to perform the same input/output behaviour. This measure is more accurate than the simple number of gates [3].

When the input to an electronic gate changes, there is a finite time delay before the change in input is seen at the output terminal. This is called the propagation delay of the gate and it differs from one gate to another. We estimate the performance of a given circuit using the worst-case delay path from input to output. The number of gate equivalent and an average propagation delay for each kind of gate were taken from [3].

Let C be a digital circuit that uses a subset or the complete set of allowed gates. The fitness function, which allows us to determine how much an evolved circuit adheres to the specified constraints, is given as follows, wherein $Gates(C)$ is a function that returns the circuit gates equivalent and function $Delay(C)$ is a function that returns the propagation delay of the circuit C based. Ω_1 and Ω_2 are the weighting coefficients that allow us to consider both area and response time to evaluate the performance of an evolved circuit. Note that the fitness function sums up a penalty ψ , which value is proportional to the number of output signal that are different from the expected ones. For implementation issue, we minimize the fitness function as $Fitness(C) = \psi + \Omega_1 Gates(C) + \Omega_2 Delay(C)$ considering the normalized values of $Gates(C)$ and $Delay(C)$ functions and the values of Ω_1 and Ω_2 equal to 0.6 and 0.4, respectively.

4 Performance Results

For each of these state machines, which can be found in [1], we evolved an optimized circuit that implements the required behaviour and compared it to the one engineered using the traditional method and to the one using the genetic programming [8]. The details of this comparison are shown in Table 2, wherein column $S/T/I/O$ provides the number of states, transitions, inputs and outputs. Note that the evolved circuits are affected by the choice of state assignment.

Table 2. Comparison of other methods vs. quantum inspired evolutionary algorithm

FSM	S/T/I/O	Number of gate-equivalent			Response time (ns)		
		Trad.	GP	Quantum	Trad.	GP	Quantum
<i>Shiftreg</i>	8/16/1/1	30	12	0	0.85	0.423	0
<i>Lion9</i>	8/16/1/1	102	33	36	2.513	0.9185	0.7690
<i>Train11</i>	9/25/2/1	153	39	43	2.945	0.8665	0.792

5 Conclusions

In this paper we exploited a quantum inspired evolutionary algorithm to synthesise the control logic used in synchronous finite state machines. We compared the circuits evolved by our algorithm with those obtained using the traditional method, i.e. through Karnaugh and transition maps as well as with GP. The state machine used as benchmarks are well known and of different sizes. Our evolutionary synthesiser always obtains better control logic either in terms of hardware area required to implement the circuit or response time. We are still synthesising some other benchmarks to validate this conclusion.

References

1. Collaborative Benchmarking and Experimental Algorithmics Lab (January 2008), <http://www.cbl.ncsu.edu:16080/benchmarks/LGSynth89/fsmexamples/>
2. Akbarzadeh-T, M.-R., Khorsand, A.-R.: Quantum Gate Optimization in a Meta-Level Genetic Quantum Algorithm. In: Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, Piscataway, NJ, USA, vol. 4, pp. 3055–3062. IEEE Press, Los Alamitos (2005)
3. Ercegovac, M., Lang, T., Moreno, J.: Introduction to Digital Systems. John Wiley and Sons, Inc., Chichester (1999)
4. Han, K.-H., Kim, J.-H.: Quantum-Inspired Evolutionary Algorithm for a Class of Combinatorial Optimization. IEEE Transactions on Evolutionary Computation 6(6), 580–593 (2002)
5. Hey, T.: Quantum computing: an introduction. Computing Control Engineering Journal 10(3), 105–112 (1999)
6. Hinterding, R.: Representation, Constraint Satisfaction and the Knapsack Problem. In: Proceedings of the Congress on Evolutionary Computation, Piscataway, NJ, USA, vol. 2, pp. 1286–1292. IEEE Press, Los Alamitos (1999)
7. Narayanan, A.: Quantum computing for beginners. In: Proceedings of the Congress on Evolutionary Computation, Piscataway, NJ, USA, vol. 3, pp. 2231–2238. IEEE Press, Los Alamitos (1999)
8. Nedjah, N., Mourelle, L.M.: Evolutionary Synthesis of Synchronous Finite State Machines. In: Evolutionary Synthesis of Synchronous Finite State Machines, 1st edn., pp. 103–128. Springer, Berlin (2004)
9. Zhang, G.-x., et al.: Novel Quantum Genetic Algorithm and Its Applications. Frontiers of Electrical and Electronic Engineering in China 1(1), 31–36 (2006)

A New Adaptive Strategy for Pruning and Adding Hidden Neurons during Training Artificial Neural Networks

Md. Monirul Islam^{1,2}, Md. Abdus Sattar¹, Md. Faijul Amin²,
and Kazuyuki Murase²

¹ Department of Computer Science and Engineering, Bangladesh University of Engineering and Technology, Dhaka 1000, Bangladesh

² Department of Human and Artificial Intelligence Systems, Graduate School of Engineering, University of Fukui, 3-9-1 Bunkyo, Fukui 910-8507, Japan

Abstract. This paper presents a new strategy in designing artificial neural networks. We call this strategy as *adaptive merging and growing strategy* (AMGS). Unlike most previous strategies on designing ANNs, AMGS puts emphasis on autonomous functioning in the design process. The new strategy reduces or increases an ANN size during training based on the learning ability of hidden neurons and the training progress of the ANN, respectively. It merges correlated hidden neurons to reduce the network size, while it splits existing hidden neuron to increase the network size. AMGS has been tested on designing ANNs for five benchmark classification problems, including Australian credit card assessment, diabetes, heart, iris, and thyroid problems. The experimental results show that the proposed strategy can design compact ANNs with good generalization ability.

Keywords: Artificial neural network design, merging neuron, splitting neuron, and generalization ability.

1 Introduction

The automated design of artificial neural networks (ANNs) is an important issue for any learning task. A too large architecture may over-fit the training data owing to its excess information processing capability. On the other hand, a too small architecture may under-fit the training data owing to its limited information processing capability. Both over-fitting and under-fitting cause bad generalization, a desirable and important property of ANNs. It is therefore necessary to design ANNs automatically so that they can solve different problems efficiently.

There have been many attempts to design ANNs automatically, such as various evolutionary and non-evolutionary algorithms (see the review papers [1]-[3]). The important parameters of any design algorithms are the consideration of generalization ability and of training time [4]. However, both parameters are important in many application areas; improving the one at the expense of the other

becomes a crucial decision. The main difficulty of using evolutionary algorithms in designing ANNs is that they are quite demanding in both time and user-defined parameters. In contrast, non-evolutionary, such as constructive, pruning and constructive-pruning, strategies require much smaller amounts of time and user-defined parameters, but they use a greedy approach in designing ANNs. Thus the design process of these strategies may trap into *architectural local optima* resulting in poor generalization.

This paper presents a non-greedy but non-evolutionary strategy in designing ANNs. This new strategy is called adaptive merging and growing strategy (AMGS). It merges and adds hidden neurons repeatedly or alternatively during the training process of an ANN. The decision when to merge or add hidden neurons is completely dependent on the improvement of hidden neurons' learning ability or the training progress of ANNs. It is argued in this paper that such an adaptive strategy is better than a predefined greedy strategy. AMGS's emphasis on using an adaptive strategy can avoid the architectural local optima problem in designing ANNs.

The rest of this paper is organized as follows. Section 2 describes AMGS in detail and gives motivations and ideas behind various design choices. Section 3 presents experimental results on AMGS and some discussions. Finally, Section 4 concludes with a summary of the paper and few remarks.

2 AMGS

The idea behind AMGS is to put more emphasis on adaptive strategy and reducing retraining epochs in the design process of ANNs. The adaptive strategy is better suited due to its ability to cope with different conditions that may arise at different stages during the design process of ANNs. This strategy also has less chance to trap into architectural local optima, a common problem suffered by a greedy strategy. The reduction of retraining epochs is suitable for undermining the effect of over training, which has a detrimental effect on the generalization ability of ANNs.

The major steps of AMGS can be explained by the following steps.

- Step 1) Create an initial ANN architecture with three layers. The number of neurons in the input and output layers is the same as the number of inputs and outputs of a given problem, respectively. The number of neurons, M , in the hidden layer is generated at random. Initialize all connection weights of the network randomly within a small range.
- Step 2) Initialize the counter $\mu_i = 0, i = 1, 2, \dots, M$. It estimates the number of epochs for which a hidden neuron is trained so far. This estimation is used for measuring the significance of hidden neurons in the network.
- Step 3) Partially train the network on the training set for a fixed number of training epochs using BP. The number of epochs, τ , is specified by the user.

- Step 4) Increment the counter $\mu_i = \mu_i + \tau, i = 1, 2, \dots, N$. Here N is the number of hidden neurons in the existing network architecture. Initially, the value of N and M is same.
- Step 5) Compute the error of the network on the validation set. If the termination criterion is satisfied, stop the training process. Otherwise, continue.
- Step 6) Compute the significance of each hidden neuron $\eta_i, i = 1, 2, \dots, N$, using the following equation.

$$\eta_i = \frac{\sigma_i}{\sqrt[3]{\mu_i}}, \quad (1)$$

where η_i and σ_i are the significance and standard deviation of the hidden neuron h_i , respectively. In AMGS, the standard deviation is computed based on a hidden neuron's output over the examples in the training set.

- Step 7) If the significance of one or more hidden neurons is less than a predefined amount, select those neurons for merging and continue. Otherwise, go to the Step 11). The significance threshold, η_{th} , is a parameter specified by the user.
- Step 8) Compute the correlation between the selected hidden neuron(s) and other hidden neurons in the network. Like standard deviation, the correlation is also computed based on the output of hidden neurons over the examples in the training set.
- Step 9) Merge one or more selected hidden neurons, if their correlated counterparts are found. The maximum number of hidden neurons that can be merged is $N/2$ for an ANN consisted of N neurons because a pair of neurons is needed for merging. Our algorithm, AMGS, merges a selected hidden neuron with an unselected neuron in the network if the correlation between these neurons is greater than a predefined correlation threshold C_{th} .
- Step 10) Retrain the modified network, which is obtained after merging operation, until the previous error level has been reached. If it is able to reach the previous error level, update the epoch counter η using Eq.(2) and go the Step 5). Otherwise, restore the previous network and continue.

$$\mu_i = \mu_i + \tau_r, i = 1, 2, \dots, P. \quad (2)$$

Here P is the number of hidden neurons in the existing network, and it is less than N because a merging operation prunes two neurons and adds one neuron. τ_r is the number of epochs for which the modified network is retrained after merging neurons.

- Step 11) Check the neuron addition criterion. If it is satisfied, continue. Otherwise, go to the Step 3) for further training. It is here assumed that since the merging criterion is not satisfied or the merging operation is found unsuccessful and the neuron addition criterion is not satisfied, the performance of the network can be only improved by training.

Step 12) Add one neuron to the hidden layer by splitting an existing neuron. The splitting produces two neurons from one neuron. Set the counter of two new neurons as $\mu_i/2$. Here μ_i is the counter of the neuron that is used for splitting. Go to the Step 3) for training the modified architecture.

It is now clear that AMGS does not guide the architecture determination process in a predefined and fixed way. Although the strategy used in AMGS seems to be a bit complex, its essence is the use of an adaptive search strategy with three components: neuron merging, neuron addition and termination criterion based on validation error. Details about each component are given in the following sections.

A. Neuron Merging

The margining operation used in AMGS consists of two steps. In the first step, the significance of each hidden neuron in an ANN is computed using Eq. (II). In the second step, a less significant hidden neuron is merged with a more significant and correlated hidden neuron. If h_a is a less significant neuron and h_b is a more significant neuron but maintains high correlation with the less significant one, AMGS merges these two neurons and produces one neuron h_c . The weights of h_c are assigned as

$$w_{ci} = \frac{w_{ai} + w_{bi}}{2}, \quad i = 1, 2, \dots, m \quad (3)$$

$$w_{jc} = w_{ja} + w_{jb}, \quad j = 1, 2, \dots, n \quad (4)$$

where m and n are the number of neurons in the input and output layers, respectively. w_{ai} and w_{bi} are the i -th input connection weights of h_a and h_b , respectively, while w_{ja} and w_{jb} are their j -th output connection weights, respectively. w_{ci} and w_{jc} are the i -th input and j -th output connection weights of h_c , respectively. Since the connection weights of h_c are obtained by combining those of h_a and h_b , it can be easily proven that the effect h_c to the ANN is almost same as the combined effect of h_a and h_b .

B. Neuron Addition

Unlike most constructive algorithms, AMGS adds hidden neurons by splitting existing hidden neurons in the network. The process of a neuron splitting is called ‘‘cell division’’ [4]. Two neurons created by splitting an existing neuron have contained the same number of connections as the parent neuron. The weights of the new neurons are calculated according to Odri *et al.* [4]:

$$w^1 = (1 + \alpha)w \quad (5)$$

$$w^2 = -\alpha w \quad (6)$$

where w represents the weight vector of the existing neuron, and w^1 and w^2 are the weight vectors of the new neurons. α is a mutation parameter whose value

may be either fixed or random. AMGS adds one neuron to the hidden layer of the ANN when its training error has stopped decreasing by a threshold, ϵ_1 , after a certain number of training epochs τ . This criterion can be described by the following equation.

$$E_t(k) - E_t(k + \tau) \leq \epsilon_1, \quad k = \tau, 2\tau, 3\tau, \dots \quad (7)$$

Here $E_t(k)$ and $E_t(k + \tau)$ are the training error at epochs k and $k + \tau$, respectively. This simple addition criterion is used widely in many constructive algorithms. AMGS tests the addition criterion after every τ epochs if the merging criterion is not satisfied or the execution of merging operation is found unsuccessful.

C. Termination Criterion

The training error of an ANN reduces as its training process progresses. However, at some point, usually in the later stages of training, the ANN may start to take advantage of idiosyncrasies in the training data. One common approach to avoid such an over-fitting is to estimate the validation error during training and stop the training process when the validation error begins to increase.

AMGS uses a very simple criterion that terminates the training process of an ANN when its validation error increases by a certain amount with respect to a minimum validation error. The criterion can be described as

$$E_{mv}(\tau) - E_v(\tau) \geq \epsilon_2. \quad (8)$$

Here $E_{mv}(\tau)$ is the minimum validation error up to training epochs τ and $E_v(\tau)$ is the validation error at epoch τ . ϵ_2 is a threshold parameter specified by the user. The termination criterion is tested after every τ epochs or when the merging operation is found successful. AMGS terminates the training process of an ANN when its validation error has increased by an amount ϵ_2 from its minimum value.

3 Experimental Studies

This section presents AMGS's performance on Australian credit card assessment, diabetes, gene, glass, heart disease, iris and thyroid problems. These benchmark classification problems have been the subject of many studies in ANNs and machine learning. The characteristics of these problems are summarized in Table [1](#), which show a considerable diversity in the number of examples, attributes and classes. The detail description of these problems can be obtained from UCI Machine Learning Repository.

Two sets of experiments were carried out. In the first of experiments, AMGS was applied to five classification problems. To observe the effect of merging and splitting, the second set of experiments was carried out. The setup of this experiments was exactly the same as those used in the first set of experiments. The only difference was that AMGS did not use here merging and splitting for pruning and adding hidden neurons, respectively. Rather, AMGS pruned neurons

Table 1. Characteristics of five benchmark classification problems

Problem	Number of				
	input attributes	output classes	training examples	validation examples	testing examples
Card	51	2	345	173	172
Diabetes	8	2	384	192	192
Heart	35	2	460	230	230
Iris	4	3	75	38	37
Thyroid	21	3	3600	1800	1800

directly and added new neurons with random initial weights. This variant of AMGS is referred to as adaptive pruning and growing strategy (APGS).

A. Results

Table 2 shows the performance of AMGS and APGS over 50 independent runs. The testing error rate (TER) refers to the percentage of wrong classifications produced by ANNs on the testing set. The number of epochs refers to the total number of training cycles required in designing final ANNs from initial ANNs.

It is clear that both AMGS and APGS could produce compact ANNs with good generalization ability by spending a reasonable amount of training epochs. However, the performance of AMGS was found better than that of APGS. For example, in terms of average results, AMGS took on 390.7 epochs for producing ANNs with 4.14 hidden neurons for the diabetes problem. APGS, on the other hand took on 420.7 epochs for producing ANNs with 5.37 hidden neurons for the same problem. It is seen from section 2 that AMGS merges correlated hidden neurons instead of pruning neurons directly. The merging operation not only reduces the size of an ANN as does pruning, but also reduces correlation among hidden neurons in the ANN. When hidden neurons are less correlated, they process less redundant information. This may be the main reason that ANNs produced by AMGS had a small number of hidden neurons. Furthermore, AMGS merges hidden neurons in such a way that one neuron produced by merging two neurons have nearly the same effect. Neurons were added in AMGS by splitting existing neurons in an ANN. As a result, the new neuron gets some information about the problem from the parent neuron. These two techniques contribute for requiring a small number of epochs in designing ANNs by AMGS.

The small value of these two parameters, i.e., number of hidden neurons and epochs, is generally considered an important aspect for obtaining good generalization ability. This may be the main reason that the TER of ANNs produced by AMGS was found better compared to that of APGS. In general, all the above examples illustrated the same point, i.e., the positive effect of using merging and splitting operation in designing ANNs.

B. Comparison

This section presents the comparison of AMGS with other related strategies in designing ANNs. We implemented basic constructive strategy (BCS), basic

Table 2. Performance of AMGS and APGS on five benchmark classification problems. All results were averaged over 50 independent runs. TER in the table indicates testing error rate.

Problem	Performance of AMGS			Performance of APGS		
	No. of	TER		No. of	TER	
	hidden neurons	epochs		hidden neurons	epochs	
Card	1.66	131.8	12.67	1.78	143.3	13.02
Diabetes	4.14	390.7	21.97	5.36	420.7	23.22
Heart	2.56	130.6	18.87	3.02	152.2	19.65
Iris	2.62	165.3	1.89	2.86	172.4	3.28
Thyroid	5.60	630.1	2.44	6.72	670.8	2.86

Table 3. Comparison between AMGS, basic constructive strategy (BCS), basic pruning strategy (BPS) and basic constructive-pruning strategy (BCPS) on three classification problems based on the number of hidden neurons, epochs and testing error rate (TER). All results were averaged over 50 independent runs.

Algorithm	No. of hidden neurons			No. of epochs			TER		
	Diabetes	Heart	Iris	Diabetes	Heart	Iris	Diabetes	Heart	Iris
AMGS	4.14	2.56	2.62	390.7	130.6	165.3	21.97	18.87	1.89
BCA	5.96	3.42	2.98	467.5	173.4	188.9	26.04	20.34	2.71
BPA	5.56	3.12	2.88	409.1	161.4	175.9	26.25	19.93	1.84
BCPA	5.80	3.26	2.88	501.3	190.7	201.6	26.22	20.43	2.71

Table 4. Comparison between AMGS, OBD [5], OBS [6], VNP [7] and OMNN [8] on the diabetes and iris problems. The results of AMGS was averaged over 50 independent runs, while they were averaged over 30 independent runs for OMNN. The results of OBD, OBS and VNP were not mentioned whether they were average or the best results. TER in the table indicates testing error rate.

Problem		AMGS	OBD	OBS	VNP	OMNN
Diabetes	No. of hidden neurons	4.14	16.0	26.0	8.0	4.53
	TER	21.97	31.40	34.60	30.90	25.87
Iris	No. of hidden neurons	2.62	4.0	4.0	2.0	2.65
	TER	1.89	2.00	2.00	2.30	4.61

pruning strategy (BPS) and basic constructive-pruning strategy (BCPS) for our base line comparisons. The implementation of these three strategies gives us an opportunity to make statistical comparison. Table 3 presents the comparison of AMGS with three other algorithms. It is observed that AMGS is better than other three strategies for the diabetes and heart problems. The *t-test* indicates that the significance of the performance difference. However, BPS performed better than AMGS for iris problem with respect to TER. The *t-test* indicates

that the performance difference is not significant. It is interesting to compare the performance AMGS with state-of-art algorithms. We therefore compared the results of AMGS with those of OBD [5], OBS [6], VNP [7], and OMNN [8]. The VNP, OBD and OBS used a pruning strategy, while OMNN used a hybrid simulated annealing and tabu search methods in designing ANN architectures. The better performance of AMGS over these state-of-art algorithms is clear from Table 4.

4 Conclusions

The generalization ability of ANNs is greatly dependent on their architectures. Although a number of strategies exist in designing ANNs, most existing strategies use a kind of greedy technique or use many user-specified parameters. This paper describes a new strategy, AMGS, in designing ANNs. The idea behind AMGS is to put more emphasis on an adaptive strategy and reducing retraining epochs in the design process of ANNs. Two techniques, pruning by merging and adding by splitting, are employed in reducing retraining epochs. The experimental results illustrate the effects of these operations (Table 2). The comparison of AMGS with other algorithms indicates the superiority of the proposed approach. One of the future works would be the use of different significance criterion in the merging operation. In addition, it would be interesting to study how well AMGS would perform on regression problems.

Acknowledgement. MMI is currently a Visiting Associate Professor at University of Fukui supported by the Fellowship from Japanese Society for Promotion of Science (JSPS). This work was in part supported by grants to KM from JSPS, Yazaki Memorial Foundation for Science and Technology, and University of Fukui.

References

1. Kwok, T.Y., Yeung, D.Y.: Constructive algorithms for structure learning in feedforward neural networks for regression problems. *IEEE Transactions on Neural Networks* 8, 630–645 (1997)
2. Reed, R.: Pruning algorithms - a survey. *IEEE Transactions on Neural Networks* 4, 740–747 (1993)
3. Schaffer, J.D., Whitely, D., Eshelman, L.J.: Combinations of genetic algorithms and neural networks- a survey of the state of the art. In: Whitely, D., Schaffer, J.D. (eds.) *International Workshop of Genetic Algorithms and Neural Networks*, pp. 1–37. IEEE Computer Society Press, Los Alamitos (1992)
4. Odri, S.V., Petrovacki, D.P., Krstonosic, G.A.: Evolutional development of a multi-level neural network. *Neural Networks* 6, 583–595 (1993)
5. LeCun, Y., Denker, J.S., Solla, S.A.: Optimal brain damage. In: Touretzky, D.S. (ed.) *Advances in Neural Information Processing Systems*, vol. 2, pp. 598–605. Morgan Kaufmann, San Francisco (1990)

6. Hassibi, B., Stork, D.G.: Second-order derivatives for network pruning: optimal brain surgeon. In: Lee, C., Hanson, S., Cowan, J. (eds.) *Advances in Neural Information Processing Systems*, vol. 5, pp. 164–171. Morgan Kaufmann, San Mateo (1993)
7. Engelbrecht, A.P.: A new pruning heuristic based on variance analysis of sensitivity information. *IEEE Transaction on Neural Networks* 12, 1386–1399 (2001)
8. Ludermir, T.B., Yamazaki, A., Zanchettin, C.: An optimization methodology for neural network weights and architectures. *IEEE Transactions on Neural Networks* 17, 1452–1459 (2006)

Using Kullback-Leibler Distance in Determining the Classes for the Heart Sound Signal Classification

Yong-Joo Chung

Department of Electronics, Keimyung University
Daegu, S. Korea

Abstract. Many research efforts have been done on the automatic classification of heart sound signals to support clinicians in heart sound diagnosis. Recently, hidden Markov models (HMMs) have been used quite successfully in the automatic classification of the heart sound signal. However, in the classification using HMMs, there are so many heart sound signal types that it is not reasonable to assign a new class to each of them. In this paper, rather than constructing an HMM for each signal type, we propose to build an HMM for a set of acoustically-similar signal types. To define the classes, we use the KL (Kullback-Leibler) distance between different signal types to determine if they should belong to the same class. From the classification experiments on the heart sound data consisting of 25 different types of signals, the proposed method proved to be quite efficient in determining the optimal set of classes.

1 Introduction

Heart auscultation is important in the diagnosis of heart diseases. Although there are some advanced techniques such as the echocardiography and the MRI, it is still widely used in the diagnosis of the heart disease because of its relatively low cost and easy accessibility. However, detecting symptoms and making diagnosis from hearing the heart sound require a skill that takes years of experience in the field.

A machine-aided diagnosis system for the heart sound signal would be very useful for assisting the clinicians to make better diagnosis of the heart disease. With the recent developments of the digital signal processing techniques, artificial neural networks (ANNs) have been widely used as the automatic classification method for the heart sound signals [1-5]. Recently, the HMM has also shown to be very effective in modeling the heart sound signal [6-7]. The highly dynamic and non-stationary nature of the heart sound signal makes it appropriate to model the signal with the HMM. In a recent study [8], they found that the HMM works much better than the ANN in classifying the heart sound signals corresponding to 10 different kinds of heart diseases. The superior performance of the HMM may come from its proven excellence in modeling non-stationary time-sequential input patterns compared with the ANN. There are many different heart sound signal types which are characterized by their unique signal shapes and spectral characteristics. Although the signal types are mainly affected by the kind of heart diseases, we can easily find cases where quite different signal types are generated from the same heart disease. From this relationship between the heart

sound signal types and the heart diseases, there is the problem of class determination in classifying the heart sound signals using HMMs. In a simple thought, one may assign a class to each type of heart sound signals. However, such an assignment will increase the number of classes indefinitely as there are too many signal types. And such a large number of classes will increase the confusability between classes and consequently result in poor classification accuracy. Also one may consider to assign the various signal types from the same heart disease to a class. However, such an approach will reduce the discrimination between classes due to poor modeling.

Although it is assumed that the general type of the heart sound signal consists of 4 components, namely, S1, systole, S2 and diastole, there are many variations in the characteristics of the heart sound signal that result in a number of signal types. The kind of the heart diseases mainly determines the signal type. However, various factors such as the severeness of the diseases, the conditions of the patient and the locations of the signal measurement also contribute to determine the type of the signal. And the signal types are distinguished from each other by the severeness of the murmurs and clicks that exist in the systole and diastole regions of the heart sound signal.

Instead of assigning a new class to each type of the heart sound signal, we propose to assign a class for a set of signal types that are acoustically-similar. In this manner, we reduce the number of classifiers that need to be constructed. Further, using this approach it is possible to determine if any unseen type of signal to appear later will be assigned to the existing class or to a new class. To define the classes for the signal types, we use the KL (Kullback-Leibler) distance [9] between different signal types to determine if they should belong to the same class. In the next section, we will explain the method how to construct classifiers using HMMs and the proposed method of determining classes based on KL distances is explained. In section 3, we show experimental results which demonstrate the feasibility of the proposed method and finally, we make conclusion in section 4.

2 Methods

2.1 Classification of Heart Sound Signals Using HMMs

A four state left-to-right HMM for a cycle of the heart sound signal is shown in Fig. 1 in line with the four components of the heart sound signal, namely S1, systole, S2 and

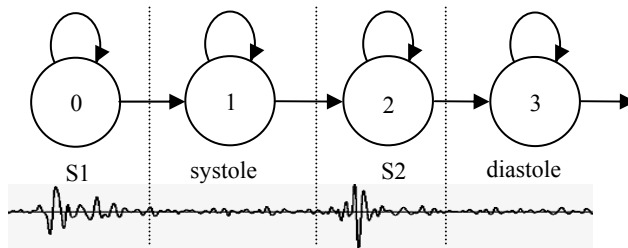


Fig. 1. An HMM for a cycle of the heart sound signal

diastole [7][8]. The number of states in the HMM is usually determined based on the nature of the signal being modeled. Each state of the HMM in Fig. 1 is assigned to a component of the heart sound signal because the signal characteristics in each component may be thought to be homogeneous. In [8], they found that the 4-state left-to-right HMM was sufficient to model a cycle of the heart sound signal. The spectral variability in each state is modeled using multiple mixtures of multivariate Gaussian distributions.

Given the observation $\mathbf{y}(t)$, the output probability distribution in the state j is given by

$$b_j(\mathbf{y}(t)) = \sum_{m=1}^M c_{jm} N(\mathbf{y}(t); \boldsymbol{\mu}_{jm}, \boldsymbol{\Sigma}_{jm}) \quad (1)$$

where $N(\mathbf{y}(t); \boldsymbol{\mu}_{jm}, \boldsymbol{\Sigma}_{jm})$ is a multivariate Gaussian distribution, with mean vector $\boldsymbol{\mu}_{jm}$ and covariance matrix $\boldsymbol{\Sigma}_{jm}$, each mixture component having an associated weight c_{jm} . Also, the transition from the state i to j is controlled by the transition probability as follows.

$$a_{ij} = P(j|i) \quad (2)$$

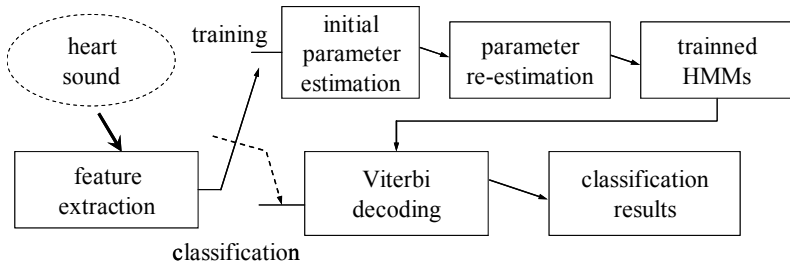


Fig. 2. The procedure of classifying the input heart sound signal using the trained HMMs

In Fig. 2, we show the procedure of classifying the heart sound signal using the trained HMMs. An HMM is constructed for each class of the heart sound signal using the training data corresponding to the class. The HMM parameters are estimated during the training procedure. For the initial parameter estimation, every cycle of the heart sound signal is manually segmented into 4 regions giving the statistical information corresponding to each element [7]. And the initial HMM parameters are re-estimated using the Baum-Welch algorithm for the maximum likelihood (ML) parameter estimation until some convergence criterion is satisfied. In classification, the Viterbi decoding is applied to find the class (HMM) which gives the best likelihood score among the trained HMMs given the input heart sound signal. The feature vectors used are 18-th order mel-frequency filter bank outputs derived from the fast Fourier transform (FFT).

2.2 Determining the Classes

Heart sound signals have various signal types depending on the cause of the signal generation. Mainly, the kind of the heart disease associated with the heart sound determines the signal type. However, various factors such as the degree of the diseases, the status of the patient and the positions of the signal measurement also contribute to determine the type of the signal. The differences between the signal types are observed in the magnitude and locations of the murmurs and clicks that usually exist in the systole and diastole regions of the heart sound signal. As there are so many signal types in the heart sound, it is not reasonable to assign a new class for each of the signal types. Such an assignment of classes will result in insufficient training of the HMM parameters and increase the confusability between classes and consequently, may lead to lower classification accuracy.

In this paper, we used 25 types of heart sound signals as shown in Table 1. The names of the signal types and the numbers of the data are also shown. The front part on the name represents the kind of disease associated with the signal type. As shown in the table, there may be several signal types from the same kind of heart disease. For example, AR_c3t2n1, AR_c3t5n1 and AR_c3t7n1 all correspond to the same kind of disease(AR). In Fig. 3, we show examples of the heart sound signal types related with the AR. As shown in the figure, the signal types show quite different characteristics although they come from the same kind of disease. From this fact, we can see that it is not reasonable to assign the signal types to the same class just because they come from the same disease. Some method to determine the optimal set of classes will be necessary.

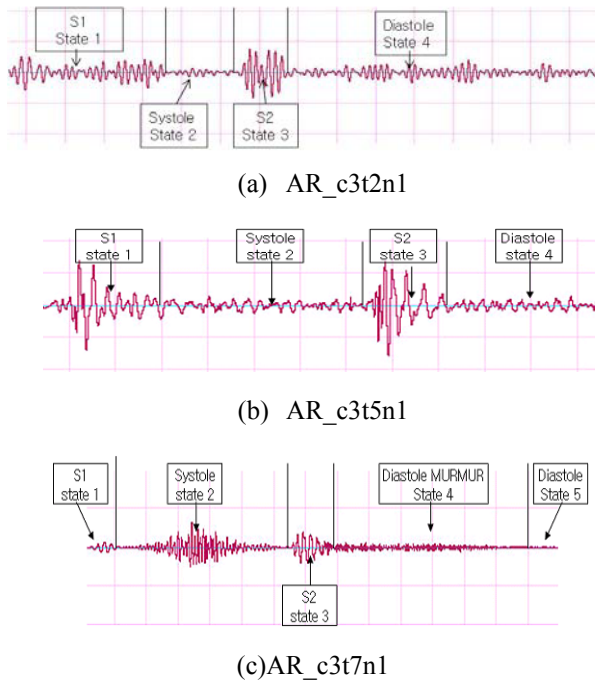


Fig. 3. Examples of the heart sound signal types related with the AR

Table 1. The various types of heart sound signals and their associated heart diseases in the classification experiment

Kind of Diseases	Signal Types	Number of Data	Kind of Diseases	Signal Types	Number of Data
NS	NS	15	MR	MR_hsmur	13
IM	IM_2lis	14		MR_absent	18
	IM_4lis	14		MR_md mur	11
AR	AR_c3t2n1	21	MS	MS_presys	20
	AR_c3t3n1	16		MS_af	19
	AR_c3t5n1	15		MS_rapid	12
	AR_c3t7n1	16		MS_ph	12
AS	AS_msmur	25	MVP	MVP_lsmur	20
	AS_apex	19		MVP_msc	15
	AS_ar	20		MVP_multi	15
	AS_prebeat	10	TR	TR_hsmur	17
	AS_cbar	15		TR_ph	42
CA	CA	20			

There are totally 9 kinds of heart diseases associated with the signal types used in this paper: NS (normal sound), IM (innocent murmur), AR (Aortic Regurgitation), AS (Aortic Stenosis), CA (Coarctation of the Aorta), MR (Mitral Regurgitation), MS (Mitral Stenosis), MVP (Mitral Valve Prolapse) and TR (Tricuspid Regurgitation). In the proposed method of determining the classes, we consider all the signal types associated with the same disease as the candidates to form the same class. Signal types from other kinds of diseases are excluded from the candidates, because the main objective of the classification is to recognize the kind of diseases of the input heart sound signal.

The class determination process proceeds as follows. We first consider an initial signal type T1 and a second type T2 from the same kind of heart disease. Our objective is to determine if the signal type T1 and T2 should be assigned as members of the same class. To determine the similarity of the signal type T1 and T2, we utilize the Kullback-Leibler (KL) distance. Signal types T1 and T2 are deemed associated with the same class if the KL distances between the corresponding states of them are all less than a threshold T. There is flexibility in the choice of the threshold T, the smaller T, the more closely related are the signal types in a given class. However, if T is made too small, the number of classes proliferates. In the same manner as above, all the signal type pairs within the same disease are checked if they can be clustered into the same class. If any signal type is not close to any other signal types within the same disease, it is solely assigned to a distinct class.

Once the classes consisting of a set of signal types which are acoustically similar are determined in the proposed algorithm, the HMM for each class is re-estimated using all the training data corresponding to the class.

3 Experimental Results

The heart sounds for the feasibility test experiments were taken from the clinical training audio CDs for the physicians [10]. The classification tests were performed with 434 heart sounds representing 25 different signal types. Each heart sound data consists of one cycle of the heart sound signal which is obtained by manually segmenting the original continuous heart sound signal. The signal types and their associated diseases were shown in Table 1. Initially an HMM was constructed during training for each type of the heart sound signal using the corresponding data. A single Gaussian mixture output distribution for each of a 4 state left-to-right HMM (Fig. 1) was used to model the heart sound signal. To overcome the problem of small amount of data collected, the classification test was done by the Jack-Knifing method. In this process, the HMM was trained with all the data available except the one used for testing. The process was repeated so that all the data can be used for testing.

Table 2. Classes determined as the threshold T of the KL distance is varied

9 kinds of diseases	25 classes (T=0)	19 classes (T=30)	18 classes (T=50)	15 classes (T=100)
NS	NS	NS	NS	NS
IM	IM_2lis	-	-	-
	IM_4lis			
AR	AR_c3t2n1	AR_c3t2n1	AR_c3t2n1	-
	AR_c3t5n1	AR_c3t5n1	AR_c3t5n1	
	AR_c3t7n1	AR_c3t7n1	AR_c3t7n1	
	AR_c3t3n1	AR_c3t3n1	AR_c3t3n1	AR_c3t3n1
AS	AS_prebeat	-	-	-
	AS_apex			
	AS_ar	AS_ar		
	AS_murmur	AS_murmur	AS_murmur	
	AS_cbar	AS_cbar	AS_cbar	
CA	CA	CA	CA	CA
MR	MR_hsmur	MR_hsmur	MR_hsmur	MR_hsmur
	MR_absent	-	-	-
	MR_mdmur			
MS	MS_presys	-	-	-
	MS_af			
	MS_rapid			
	MS_ph			
MVP	MVP_lsmur	MVP_lsmur	MVP_lsmur	MVP_lsmur
	MVP_msc	-	-	-
	MVP_multi			
TR	TR_hsmur	TR_hsmur	TR_hsmur	TR_hsmur
	TR_ph	TR_ph	TR_ph	TR_ph

Table 2 shows the class determined by the proposed method as the threshold T of the KL distance varies. If we assume $T=0$, every signal type constitutes a distinct class resulting in 25 classes. As shown in the table, as we increase T to 100, the number of classes dropped to 15. If we increase T to infinity, the number of classes will be 9 equal to the number of heart diseases listed in the first column of the table. The bar (-) sign in the table means that the classes in the left column have been merged into the same class although the name of the new class is not given for notational simplicity. For example, the signal type IM_2lis and IM_4lis which constitute separate classes when $T=0$ are merged together to form a new class when $T=30, 50, 100$.

Fig. 4 also shows the classification accuracy as the threshold T varies. When all the signal types from the same heart diseases are assigned to the same class by letting T equal to infinity, the classification accuracy is very low. This means that the various types of signals from the same heart disease differ too much in their characteristics to be included in the same class. The HMMs for the classes will not be modeled sharply enough to distinguish one from another. The performance was found to be unsatisfactory when every signal type had its own distinct class (25 classes) at $T=0$, because the perplexity of classification was high. The highest classification accuracy was achieved when the number of classes is 19 with $T=30$. Fig. 4 shows that the optimal set of classes can be determined by setting an appropriate threshold value T . We also compared the proposed method with a heuristic method in which the set of classes is determined based on the observed similarity in the shapes of the signal waveform. The heuristic method produced 21 classes. We can see that the proposed method works better than the heuristic method when $T=30$ although a slight performance degradation was observed when $T=50$. Despite some perturbation in performance with T , the proposed method is advantageous since it is based on the objective criterion of the KL distance proven to be efficient in measuring the similarity between two statistical models. In contrast, the heuristic method is subjective and may show large variability in classification depending on the signal types.

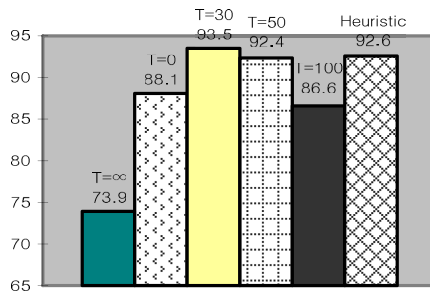


Fig. 4. Classification accuracy as the threshold T is varied

4 Conclusion

In this paper, we have proposed a method to determine the optimal set of classes in the heart sound signal classification. As there are many types of signals even from the same kind of heart diseases, it is necessary to cluster the various signal types into classes according to their similarity rather than assigning a separate class to each type of the signals. As we employ HMMs to model the heart sound signals, the KL distance which has shown to be efficient in measuring the similarity between statistical models was utilized to determine if the signal types from the same heart disease should be clustered into the same class. From the experimental results, we could see that the proposed method improved classification accuracy remarkably compared with the case when we assign a distinct class to each signal type and it also works better than the case when all the signal types in the same kind of diseases are assigned to a class. In addition, the proposed method is advantageous because it performs well compared with the heuristic method without the need to be subjective in determining the classes.

Acknowledgments. This work has been supported by The Advanced Medical Technology Cluster for Diagnosis and Prediction at KNU, which carries out one of the R&D Projects sponsored by the Korea Ministry Of Commerce, Industry and Energy.

References

- 1 Leung, T.S., White, P.R., Collis, W.B., Brown, E., Salmon, A.P.: Acoustic diagnosis of heart diseases. In: Proceedings of the 3rd international conference on acoustical and vibratory surveillance methods and diagnostic techniques, Senlis, France, pp. 389–398 (1998)
- 2 Cathers, I.: Neural Network Assisted Cardiac Auscultation. *Artif. Intell. Med.* 7, 53–66 (1995)
- 3 Bhatikar, S.R., DeGroff, C., Mahajan, R.L.: A Classifier Based on Artificial Neural Network Approach for Cardiac Auscultation in Pediatrics. *Artif. Intell. Med.* 33, 251–260 (2005)
- 4 Lippmann, R.P.: An Introduction to Computing with Neural Nets. *IEEE ASSP Magazine*, 4–22 (April 1987)
- 5 DeGroff, C., Bhatikar, S., Hertzberg, J., Shandas, R., Valdes-Cruz, L., Mahajan, R.: Artificial neural network-based method of screening heart murmur in children. *Circulation* 103, 2711–2716 (2001)
- 6 Gill, D., Intrator, N., Gavriely, N.: A Probabilistic Model for Phonocardiograms Segmentation Based on Homomorphic Filtering. In: 18th Biennial International EURASIP Conference Biosignal, pp. 87–89 (2006)
- 7 Ricke, A.D., Pavinelli, R.J., Johnson, M.T.: Automatic segmentation of heart sound signals using hidden Markov models. *Computers in Cardiology*, 953–956 (September 2005)
- 8 Chung, Y.: A Classification Approach for the Heart Sound Signals Using Hidden Markov Models, *SSPR/SPR*, pp. 375–383 (2006)
- 9 Rabiner, L.R., Wilpon, J.G., Juang, B.H.: A segmental k-means training procedure for speech recognition. *IEEE Trans. ASSP*, 2033–2045 (December 1990)
- 10 Juang, B.H., Rabiner, L.R.: A probabilistic distance measure for hidden Markov models. *AT&T Tech. J.*, 391–408 (1984)
- 11 Mason, D.: *Listening to the Heart*, Hahnemann University (2000)

A Semi-fragile Watermark Scheme Based on the Logistic Chaos Sequence and Singular Value Decomposition*

Jian Li¹, Bo Su², Shenghong Li¹, Shilin Wang², and Danhong Yao¹

¹ Electronic Engineering Department, Shanghai JiaoTong University, Shanghai, China

² School of Information Security Engineering, Shanghai JiaoTong University, Shanghai, China
lijianlj@sjtu.edu.cn

Abstract. In this paper a semi-fragile watermark scheme is proposed based on the Logistic chaotic sequence and singular value decomposition (SVD). Chaotic sequence is very sensitive to the initial value and has zero-value of cross-correlation. With these properties of chaotic sequence, the uniqueness of the watermark can be efficiently obtained. Some singular values show robustness under lossy compression and can be adopted as a good carrier of watermark. Hence SVD quantization is also employed in our algorithm. In the proposed approach, algorithm efficiency and the influence on host image quality will be discussed theoretically and experimentally. The experimental results show the high robustness against lossy compression and the place being maliciously tampered can be accurately detected and located on the protected digital images.

Keywords: Logistic chaos sequence; Singular value decomposition (SVD); Semi-fragile Watermark.

1 Introduction

Semi-fragile watermark is mainly applied to the integrity verification of image. By embedding watermark into host image, authentication can be performed by detecting the embedded fragile watermark in watermarked image. The SVD-based methods to embed watermark first appear at robust watermark field and its application in fragile watermark field is relatively rare. The first application of singular decomposition in watermark field is from the Liu's papers [2][3], however in [4][5], scholars have pointed out some drawbacks of this kind of schemes. Then more researchers proposed improved singular decomposition scheme. The widely used methods in semi-fragile watermark schemes are performed by modifying the value in space domain and transform domain. The commonly-used transform domains include DFT, DCT, DWT etc.

Considering that currently prevalent images are almost compressed, it's a challenge for semi-fragile watermark to survive under high compression. Our algorithm aims to solve the survive problem under lossy compression, as well as provide accurate location for image content tamper. The experimental results show that even in

* This research is funded by NSFC of China under grant number of 60772098/ 60702043, NCET of Ministry of Education of China under grant number of NCET-06-0393, National 863 Hi-Tech Research and development plan of China under grant number of 2007AA01Z455, and Shanghai dawn scholar Foundation in China.

high lossy compression condition, our scheme also can work effectively. We also provide theoretical analysis for influence on original after being watermarked. This paper is organized as follows: in section 2, we introduce the preliminary knowledge. The details of our scheme are presented in section 3 and the theoretical error and security analysis will be discussed in section 4. The experiment is in section 5.

2 Chaotic Sequence and Singular Value Decomposition (SVD)

2.1 Logistic Chaotic Sequence

Chaotic sequence is a kind of sequence that is neither periodic nor convergent. Moreover it is very sensitive to initial values. The process chaotic sequence represents is like random. By using Logistic mapping, we can obtain chaotic sequence. Logistic mapping is a very simple but extensively applied in many fields. The following is Logistic full mapping model: x_n is the mapping variable, let x_0 as the initial value

$$x_{n+1} = 1 - 2 \times x_n^2, x_n \in (-1, 1), x_n \neq 0 \quad (1)$$

Its probability function is:

$$\rho(x) = \frac{1}{\pi \sqrt{1-x^2}}, x \in (-1, 1), x_n \neq 0 \quad (2)$$

The statistic characteristics of chaotic sequence x_n as follow:

- (1) Mean: $\bar{x} = 0$
- (2) Autocorrelation function: $R(\tau) = 1$
- (3) Cross-correlation function: $c(m) = 0$

The mapping result is decimal with value within -1 to 1. Then using the binary quantization to get the binary sequence, the quantization function is:

$$m(x) = \begin{cases} 1, \tau \geq 0 \\ 0, \tau < 0 \end{cases} \quad (3)$$

From the above formulas, we can learn that the autocorrelation of chaotic sequence is equal to one, while the cross-correlation is zero between different sequences. In other words, if we distribute unique initial value of chaotic sequence as secret key to different owner of watermarked images, an owner just verifies his own watermarked image because different secret keys generate totally different sequences with zero value of cross-correlation. The detail of how to use chaotic sequence in our scheme will be discussed in Section 3.

2.2 Singular Value Decomposition (SVD) and Quantization

Here we propose a new kind of SVD watermark scheme. From the experiments, we have discovered some singular values are relative robustness against lossy compression such as JPEG and JPEG 2000. And those robust singular values can be utilized to embed watermark. The singular value decomposition can be expressed:

$$A = U \times S \times V^*, A \in F^{M \times N}, U \in F^{M \times M}, V \in F^{N \times N}, S \in R^{M \times N} \quad (4)$$

Here A denotes the host image, U, V is unitary matrix, S is the diagonal matrix:

$$S = \begin{bmatrix} s_1 & & \\ & \ddots & \\ & & \ddots \end{bmatrix} \quad (5)$$

Using F norm to represent the energy of the image A , the definition of F norm is:

$$\|A\|_F = \left(\sum_{i,j} |a_{i,j}|^2 \right)^{1/2} = \sqrt{\text{tr}(A^*A)} \quad (6)$$

Substituting A with formula (4):

$$\|A\|_F = \sqrt{\text{tr}(A^*A)} = \sqrt{\text{tr}((USV^*)^*(USV^*))} = \sqrt{\text{tr}(VS^*U^*USV^*)} = \sqrt{\text{tr}(V\|S\|^2V^*)} \quad (7)$$

Due to U and V are unitary matrix with inner product equal to 1, moreover S is the diagonal matrix, then:

$$\|A\|_F = \|S\| = \left(\sum_{i=1}^{\min(M,N)} s_i^2 \right)^{1/2} \quad (8)$$

We obtain that the F norm of S equal to the F norm of A . Thus the energy of an image is equal to the evolution of the sum of the diagonal element square of matrix S . In most common situation, there is one singular which is much greater than others. Thus the biggest singular value can represent the whole energy of image A , while other singular values just represent the energy of image detail. In our experiment, we discover that the variation of the biggest singular value (denoted by S_{\max}) is small under JPEG/JPEG2000 lossy compression. Fig.1 shows that the probability distribution curve of S_{\max} variation range under different compressions.

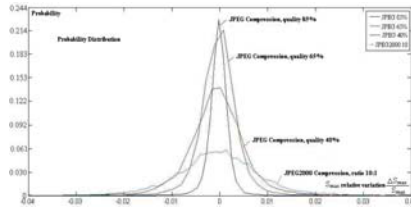


Fig. 1. The probability distribution curve of S_{\max} variation range

From the above figure, we can learn that S_{\max} meets the normal distribution. With different compression ratio, the variation range is also different. But the range of the variance is relative small from -0.02 to 0.02. Therefore we can utilize the relative stable value S_{\max} under lossy compression to embed watermark.

Before discussing the watermark embedding, the concept of quantized interval needs to be illustrated. Due to it varies in certain range, S_{\max} is divided into several quantized interval whose length is twice larger than the variation range of S_{\max} and whose terminal points represent binary bits. The quantized interval table (in Fig.2)

shows the terminal point and binary bits. If one bit of watermark needs to be embedded, we just compare S_{\max} with terminal point in the table and modify S_{\max} .

Suppose the upper limit of quantized interval equals to C that should be greater than any S_{\max} in blocks, C is divided into several interval with step $\Delta q_i = |q_i - q_{i+1}|$, where the terminal point value is $q_i, i \in (0 \sim I)$, I is the number of intervals. q_i can be allocated by different strategies such as equal distance or equal proportion. Here Δq_i should be greater than the variation range of S_{\max} two times so as to resist the lossy compression. Next paragraph will demonstrate how to use quantized interval to embed watermark bits.

S_{\max} Quantized value	The represented bit
\vdots	\vdots
q_{i+3}	1
q_{i+2}	0
q_{i+1}	1
q_i	0
\vdots	\vdots

Fig. 2. The allocation of quantized interval of singular value

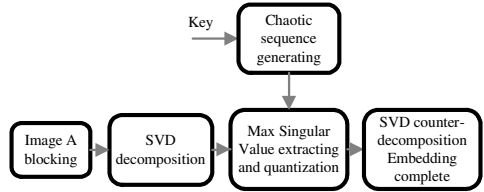


Fig. 3. The flow chart of embedding watermark

3 Algorithm Procedure Based on SINGULAR Value Decomposition

Our scheme can be applied to both grayscale and color image. The grey value of pixels for grayscale image or the luminance value Y for color image (transmitting the RGB into YUV) is the place where we embed watermark. Figure 3 illustrates how to embed watermark. The initial value is the owner’s key to generate the chaotic sequence. Host image A is transformed into SVD domain, quantizing SVD value S_{\max} to embed watermark according to the chaotic sequence. The detail of embedding process will be shown in section 3.1. When the receiver receives the watermarked image, the receiver generates a chaotic sequence which should be identical to the sequence extracted from watermarked image because both sequences are generated by same initial key. By comparing the sequences, we can locate the tampered position in image and achieve our anti-tamper goal. The extracted process will be illustrated in section 3.2.

3.1 Embedding Algorithm

Step 1: Partition the host image A into blocks represented by A_k with size 4×4 respectively, where $k = (1 \dots K)$ and K is the total number of the blocks. Then every block is applied to singular value decomposition, extracting the singular values S_{\max} from the diagonal matrix S_k (Eqn. 5).

Step 2: Use initial value as the secret key to produce chaotic sequence x_k (Eqn. 1), $k = (1 \dots K)$. One bit of sequence is in accordance with one block A_k .

Step 3: Select suitable parameters to generate quantized interval and extract $S_{k,\max}$ (the biggest singular value in S_k) by: comparing $S_{k,\max}$ with q_i in Fig.2, if $q_i \leq S_{k,\max} < q_{i+1}$, then compare the represented bit of q_i and q_{i+1} with the watermark bit x_k , if the represented bit q_i equal to x_k , substitute $S_{k,\max}$ with q_i , vice versa with q_{i+1} . The new $S_{k,\max}$ is denoted by $S_{k,\max}^*$.

Step 4: Block is applied to singular value counter-decomposition and move to next block until all blocks finish this kind of operation.

3.2 Detecting Algorithm

Step 1: The received image is first divided into blocks with size 4×4 . Then singular value decomposition is performed in every block to extract the diagonal singular value matrix \tilde{S}_k and biggest singular value $\tilde{S}_{k,\max}$.

Step 2: Use the same method to allocate quantized interval in accordance with the embedding algorithm. The biggest singular value $\tilde{S}_{k,\max}$ is compared with q_i .

If $q_i \leq \tilde{S}_{k,\max} < q_{i+1}$, then calculate the distance d_i and d_{i+1} respectively: where $d_i = \tilde{S}_{k,\max} - q_i$, $d_{i+1} = q_{i+1} - \tilde{S}_{k,\max}$. After obtained the value d_i and d_{i+1} , we decide the embedded watermark bit according to decision rule. For example: if $d_i \geq a \times d_{i+1}$ (a is a proportional factor, if $a = 1$, then this is average decision.), then take the represented bit of q_{i+1} as the watermark bit, if $d_i \leq a \times d_{i+1}$ take q_i .

Step 3: Move to next block to detect until all bits being extracted from blocks.

Step 4: Use the same secret key as in the embedding counterpart to generate chaotic sequence x_k .

Step 5: Compare of chaotic sequence x_k with the watermark bits extracted from image.

If the two corresponding bits are different, then this block has been tampered, mark it out.

4 The Error and Security Analysis

Since we just embed watermark bits in the biggest singular value S_{\max} , the influence on host image can be measured by this singular value. Here we use matrix norm and PSNR (Peak Signal Noise Ratio) to measure the quality variation on host image and the error brought about. According to equation (8), we get:

$$\|A\|_F = \|S\|_F = \left(\sum_{i=1}^{\min(M,N)} s_i^2 \right)^{1/2} \Rightarrow \quad (9)$$

$$\text{The error of the } k\text{th block: } \|\Delta A_k\|_F = \|\Delta S_k\|_F = \left(\sum_{i=1}^{\min(M,N)} \Delta s_{i,k}^2 \right)^{1/2} = \Delta S_{k,\max} = |S_{k,\max} - S_{k,\max}^*|$$

where $S_{k,\max}^*$ is the terminal value of quantized interval q_i or q_{i+1} . Moreover, quantized interval $\Delta q_i = |q_i - q_{i+1}|$ is relative enough small compared to $S_{k,\max}$. We assert that $S_{k,\max}$ is even distribution in quantized interval from q_i to q_{i+1} . $S_{k,\max} - S_{k,\max}^*$ is even distribution with mean equal to $\Delta q_i / 2$, variance equal to $\Delta q_i^2 / 12$, Therefore:

$$\begin{aligned} E(\|\Delta A_k\|_F^2) &= E(\|\Delta S_k\|_F^2) = E(|\Delta S_{k,\max}|^2) = E(|S_{k,\max} - S_{k,\max}^*|^2) \\ &= D(|S_{k,\max} - S_{k,\max}^*|) + (E|S_{k,\max} - S_{k,\max}^*|)^2 = \frac{1}{12} \Delta q_i^2 + \left(\frac{1}{2} \Delta q_i\right)^2 = \frac{1}{3} \Delta q_i^2 \end{aligned} \quad (10)$$

$E(\cdot)$ denotes mean, $D(\cdot)$ denotes variance, Δq_i is the length of quantized interval.

Therefore the average error square \bar{E}_{pixel} of single pixel of image is:

$$\bar{E}_{pixel} = \frac{1}{M * N} E(|S_{k,\max} - S_{k,\max}^*|^2) = \frac{1}{16} \times \frac{\Delta q_i^2}{3} = \frac{c^2}{48} \quad (11)$$

Here block size $M = N = 4$, quantized interval is set to the same length size. For simple discussion, $\Delta q_i = c$, c is constant value. In next section, we will discuss the comparison of the calculated result of \bar{E}_{pixel} with measured PSNR.

The security of our watermark scheme is mainly composed by two parts. One is from the high security of chaotic sequence. Due to the result from chaotic mapping is decimal, before using the chaotic sequence, the result need to be quantized. Therefore it is very difficulty to decode or decipher the chaotic sequence without the initial value of sequence or deduce the initial value from the quantized sequence. The other security point is the allocation of quantized interval. Use different strategies to allocate interval such as the adaptive strategies which is not only to enhance the security of the watermark information but also to improve the quality of the watermarked image. Therefore without knowing the details about length and terminal point of quantized interval, to recover the watermark bits is difficult.

5 Experiment Result and Discussion

As uniform quantization is adopted, the length of interval $\Delta q_i = |q_i - q_{i+1}| = 17$, LENA 512×512 grayscale image is adopted as watermark host image and PSNR is adopted to measure the quality variation on host image. Fig. 5 shows that there is no remarkable change on image quality between the original image and watermarked image and the measured value of PSNR is 40.142dB. Such result indicates that our watermark scheme has good transparency. While according to the error analysis in previous

section, we put $\Delta q_i = c = 17$ into (11), then obtain $\bar{E}_{pixel} = 289/48$. Substitute \bar{E}_{pixel} in (12) with $\bar{E}_{pixel} = 289/48$, where $f(m,n)$ is the original image pixel, $g(m,n)$ is the modified image pixel.

$$PSNR = 10 \log_{10} \frac{255^2}{\frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} [f(m,n) - g(m,n)]^2} = 10 \log_{10} \frac{255^2}{\frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} [\bar{E}_{pixel}]^2} = 40.334dB \quad (12)$$

The above value is the ideal value. In reality, substituting original value with quantized value in embedding watermark inevitably brings about decimal fraction in pixel domain which will be cast out, result in PSNR value reduction. However, the measured value PSNR is well accorded to our calculated value validates our algorithm. In Fig.4, the relationship between length of quantized interval and PSNR is shown. The smaller the interval is, the less influence the watermark has on host image.

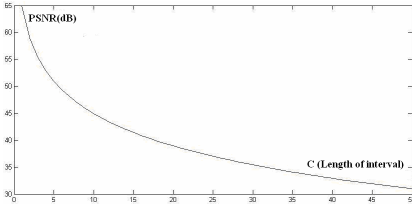


Fig. 4. The relationship between quantized interval and PSNR

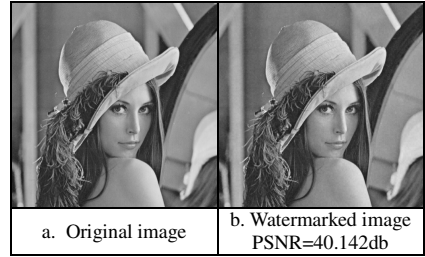


Fig. 5. The comparison experiment of watermark embedding

Currently the popular image is in compression format, such as JPEG, JPEG2000. The tampering experiment process is as follows: we modify three positions on a watermarked image, respectively, i) writing a character on the image; ii) removing an image block; iii) adding an icon irrelevant to the image. Then the watermarked image is compressed (including JPEG85%, 65%, 40% quality compression, JPEG2000 10:1, 15:1, 20:1 compression). Finally we list the detect result under various conditions. The test result shows that even JPEG compression with quality of 40%, the tampered position can still be obviously seen and accurately located. We use TAF (including the missing rate and false alarm rate) to estimate the error rate caused by detecting. TAF is defined as: $TAF = (\text{false detected bit number}) / (\text{total watermark bit number})$.

It can be observed from Figure 5 that the detected results have some noises under high compression. But these noises are separate and randomly distributed, thus can be recognized by eyes and removed by adding median filter as well. Here we propose a filtering method as follows: i) transverse one-dimension median template with a window size 3 is employed to perform filtering; ii) vertical one-dimension median template with a window size 3 is adopted to perform filtering; iii) the two filtering results are added. From Figure 6.i.j we can see that this filtering method could remove most


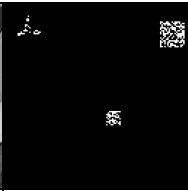



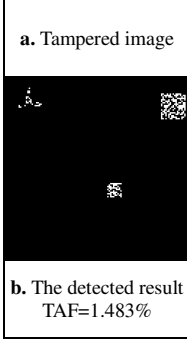

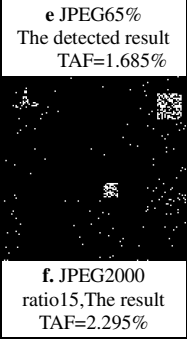
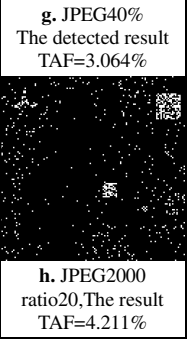
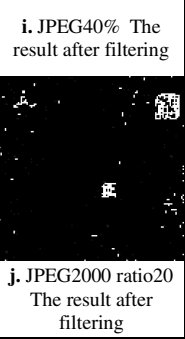
				
				

Fig. 6. The experiment result

of the noises and meanwhile remain most effective results. The concept of quantized interval is crucial to this algorithm. Modifying the interval Δq_i can control the PSNR value. Yet the interval can be dynamically decided by adaptively analyzing the histogram of host image.

6 Conclusion

In this paper, an effective algorithm is proposed to resist lossy compression such as JPEG and JPEG2000, etc. The proposed algorithm can accurately locate the tampered position of an image and manipulate the influence on host image in controllable way. Our algorithm can achieve blind detection and watermark extraction without extra information. A median filtering method is also proposed and is demonstrated to be able to effectively improve the detection quality.

References

1. Dijun, X.: A study of digital image spread watermark algorithm based on chaotic sequence. Shangdong university (2006)
2. Liu, R., Tan, T.: An SVD-based watermarking scheme for protecting rightful ownership. *IEEE Transactions on Multimedia* 4(1), 121–128 (2002)
3. Liu, R., Tan, T.: SVD Based Digital watermarking Method. *ACTA Electronica Sinica* 29(2), 1–4 (2001)
4. Rykaczewski, R.: Comments on An SVD-Based Watermarking Scheme for Protecting Rightful Ownership. *IEEE Transactions on Multimedia* 9(2), 421–423 (2007)
5. Xiao-Ping, Z., Kan, L.: Comments on An SVD-based watermarking scheme for protecting rightful Ownership. *IEEE Transactions on Multimedia* 7(3), 593–594 (2005)

Distribution Feeder Load Balancing Using Support Vector Machines

J.A. Jordaan, M.W. Siti, and A.A. Jimoh

Tshwane University of Technology
Staatsartillerie Road, Pretoria, 0001, South Africa
jordaan.jaco@gmail.com, willysiti@yahoo.com, jimohaa@tut.ac.za

Abstract. The electrical network should ensure that an adequate supply is available to meet the estimated load of the consumers in both the near and more distant future. This must of course, be done at minimum possible cost consistent with satisfactory reliability and quality of the supply. In order to avoid excessive voltage drop and minimise loss, it may be economical to install apparatus to balance or partially balance the loads. It is believed that the technology to achieve an automatic load balancing lends itself readily for the implementation of different types of algorithms for automatically rearranging the connection of consumers on the low voltage side of a feeder for optimal performance. In this paper the authors present a Support Vector Machines (SVM) implementation. The loads are first normalised and then sorted before applying the SVM to do the balancing.

1 Introduction

The distribution system technology has changed drastically, both quantitatively and qualitatively. This may be ascribed to the fact that with increase in technological development, the dependence on electric power supply has increased considerably. Consequently, while demand has increased, the need for a steady power supply with minimum power interruptions and fast fault restoration has also increased. To meet these demands, automation of the power distribution system needs to be widely adopted. All switches and circuit-breakers involved in the controlled networks are equipped with facilities for remote operation. The control interface equipment must withstand extreme climatic conditions. Also, control equipment at each location must have a dependable power source. To cope with the complexity of the distribution, the latest computer, communication, and power electronics equipment in distribution technologies are needed to be employed. The distribution automation can be defined as an integrated system concept. It includes control, monitoring and some times, decision to alter any kind of loads. The automatic distribution system provides directions for automatic reclosing of the switches and remote monitoring of the loads contributing towards phase balancing.

The distribution system will typically have a great deal of single-phase loads connected to them. Therefore distribution systems are inherently unbalanced.

The load is also very dynamic and varies with time; these factors contribute to increase difficulties in controlling the distribution voltage within certain limits. In addition to this most of the time the phases are unequally loaded and they produce undesired negative and zero sequence currents. The phase voltage and current unbalances are major factors leading to extra losses, communication interference, equipment overloading and malfunctioning of the protective relay which consequently results into service quality and operation efficiency being reduced [1]. Phase unbalance is also manifested in increased complex power unbalance, increased power loss, enhanced voltage drop, and increased neutral current.

Traditionally, to reduce the unbalance current in a feeder the connection phases of some feeders are changed manually after some field measurement and software analysis. Although in some cases this process can improve the phase current unbalance, this strategy is more time-consuming and erroneous. In this paper the use of support vector machine (SVM) based load balancing is proposed as a novel procedure to perform the feeder phase balancing. In most of the cases, the phase voltage and current unbalances can be greatly improved by suitably arranging the connection phases between the distribution transformers and a primary feeder. It is also possible to advance the phase current unbalances in every feeder segment by means of changing the connection phases [2]. The phase voltage unbalances along a feeder can also be improved in common cases by system reconfiguration, which involves the rearrangement of loads or transfer of load from heavily loaded areas to the less loaded. In the modern power distribution system, the sectionalizing switches and the tie switches for feeder reconfiguration are extensively used [1]. The authors in [3] presented the way to control the tie switches using heuristic combinatorial optimization-based method. The only disadvantage with the tie-switch control is that, in most of the cases, it makes the current and the voltage unbalances worse. The reference [4] presented the use of the neural networks to find the optimum switching option of the loads among the different phases.

The layout of the paper is as follows: in section 2 we discuss the current methods of load balancing and introduce the new proposed method of treating the load balancing problem, section 3 shows the numerical results obtained, and the paper ends with a conclusion.

2 Problem Description

2.1 Representation of the Feeder

In South Africa a distribution feeder is usually a three-phase, four-wire system. It can be a radial or open loop structure. The size of the conductor for the entire line of the feeder is the same. These feeders consist of a mixture of loads, e.g. commercial, industrial, residential, etc. Single-phase loads are fed by single-phase two-wire service, while three-phase loads are fed by three-phase four-wire service. In Fig. 1 each load can be connected through the switch selector only to one of the three phases.

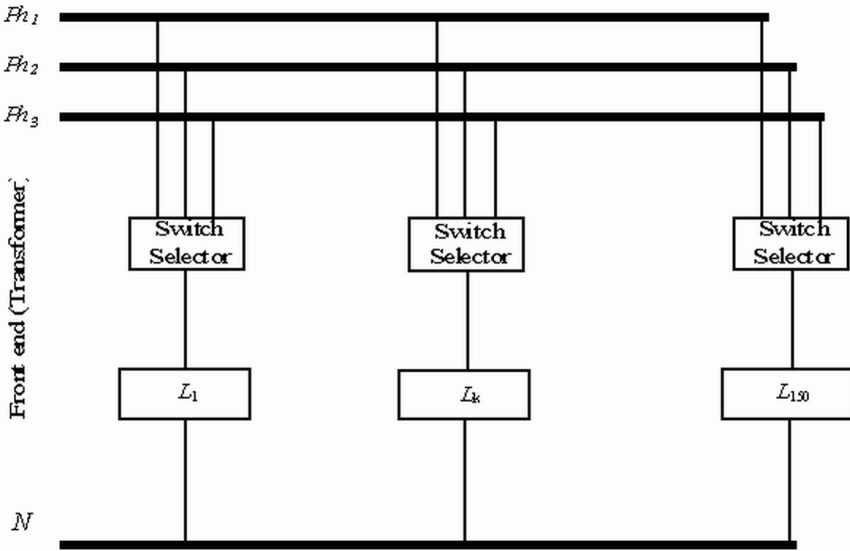


Fig. 1. Three-phase System

The daily load pattern is a function of time and of the type of customers. The resulting power system voltages at the distribution end and the points of utilization can be unbalanced due to several reasons. The reasons include the following: fundamental phase angle deviation; unequal voltages magnitude at the fundamental system frequency (under voltage and over voltages); asymmetrical transformer winding impedances [5], etc. A major cause of this unbalance is uneven distribution of single-phase loads that are continually changing across a three-phase power system. Normally the consumption of consumers connected to a feeder fluctuates, thus leading to the fluctuation of the total load connected to each phase of the feeder. This in turn implies that the degree of unbalance keeps varying. The worse the degree of unbalance the higher the voltage drop and the less reliable the feeder is.

Minimum power loss reconfiguration is aimed at by means of controllable switch-breakers installed at each of the connections on the network feeders, since both the loads and the switch-breaker status are physically distributed. In the general formulation of the phase balancing problem, the load values are the independent variables, whereas the switch-breaker statuses are the optimization variables. The objective can be fulfilled performing a control strategy in which the status of each switch-breaker depends on the total load from each feeder. In this way, the network can be optimally operated and it is not necessary to know the load in advance. For the real implementation of a control system, the following elements are necessary:

- A measurement system for real loads.
- Data system for the load data connecting to each point.

- Transmission system for sending the input signals to the switch breaker.
- The control cannot start if the above described components and system are not properly installed and in correct condition.

2.2 Current Phase Balancing Technique

Most of the township houses in South Africa on average use about three kilowatt power. The major electricity usage is for lighting and domestic works in the domestic environment. However, sudden power increase, like the use of heaters, etc., often times introduces an unknown power in the distribution system, which could damage the transformers and burn the cables, causing unbalance in the network. To balance the network, the engineers and the technicians must change the phases manually after some field measurements. The changes made to upgrade transformation in different areas affect the size of the conductor, but in most of the cases, the size of the phase conductor for the entire line of the feeder is the same. However, a number of phase conductors may be different in different sections for economic reasons. The power losses depend on the real and the reactive power flows, which are related to the real and reactive loads.

2.3 Proposed Solution

The proposed solution to the load balancing problem is based on a SVM implementation. The loads are first normalised by dividing by the maximum value and then sorted in ascending order before applying the SVM to do the balancing. In Fig. 2 an example of three sets of loads is shown. The unsorted load patterns differ greatly, while the sorted load patterns (the loads in each data set is sorted) show a similar curve. This sorting of the loads should enhance the performance of the SVM. It should be noted that the sorted load curves for each set of normalised loads do not look the same. However, since all curves start at a low value and increase to a maximum value of one, the use of SVMs is preferred above other non-linear regression methods.

For this application we will only work with 15 loads. This means we may have many sets of loads, where each set has 15 loads. The inputs to the support vector machine are the different sets of 15 load currents at each of the consumers and the outputs indicate to which phase each load should be connected. The output of the network is in the range $\{1, 2, 3\}$ for each load, i.e., which switch (to the specific phase) should be closed for that specific load. In this case the SVM will have 15 inputs and 15 outputs.

3 Numerical Results

For this experiment we tested many linear and non-linear SVM regression methods. For the results we show only the non-linear Radial Basis function (RBF) kernel SVM. For the implementation we used MATLAB [6] and the Least Squares Support Vector Machines toolbox from [7]. The RBF kernel is given by

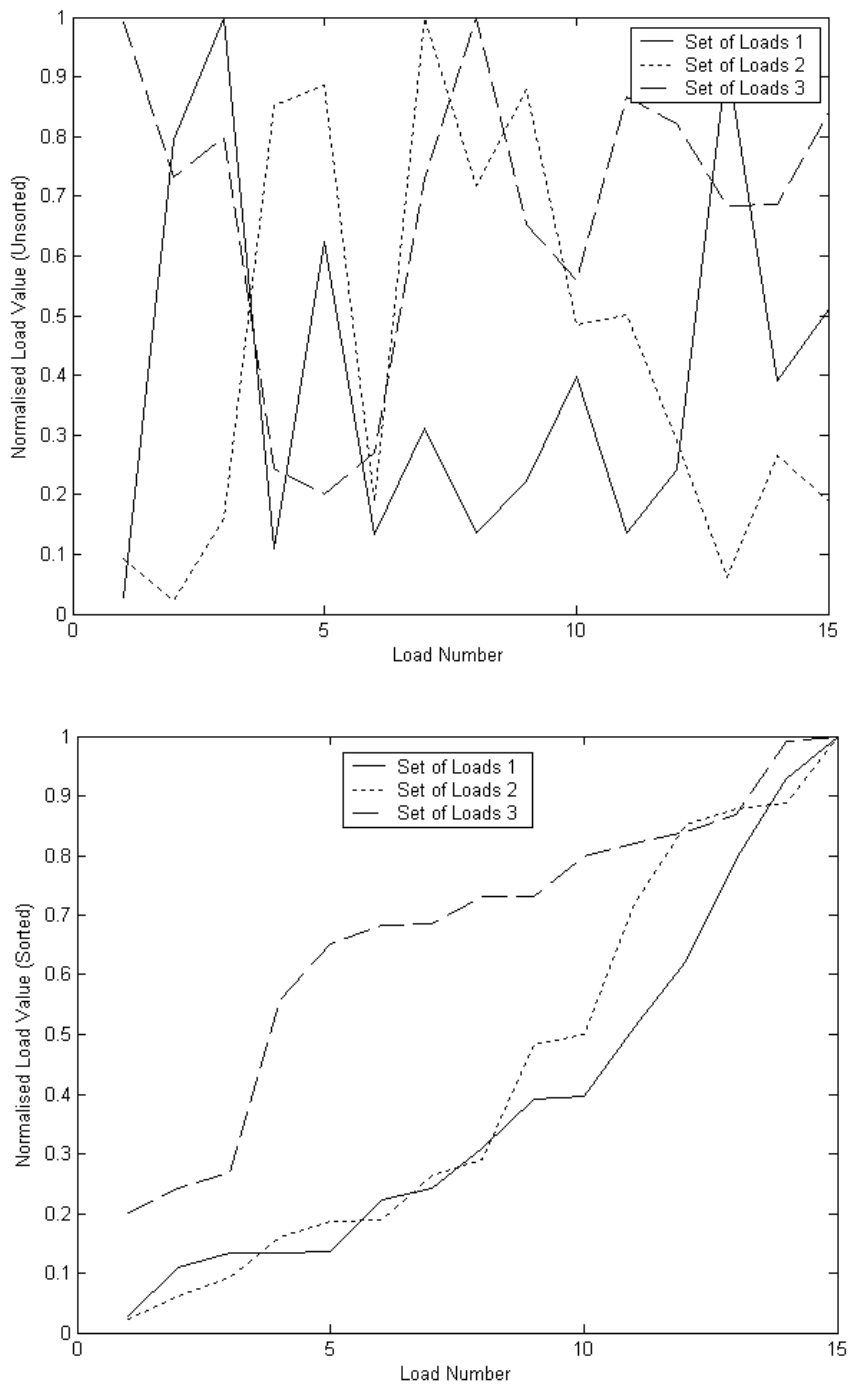
**Fig. 2.** Sorted Loads vs Unsorted Loads

Table 1. Results of the Two SVMs

Parameter	Unsorted	Sorted
Percentage of the loads where this method gave best balancing	28.8%	71.2%
U_{1-2}	33699 A	25292 A
U_{1-3}	38945 A	24551 A
U_{2-3}	30636 A	16923 A
U_T	103280 A	66766 A

Table 2. Results of the SVM and the Heuristic Method

Parameter	Heuristic	Sorted
Percentage of the loads where this method gave best balancing	27.6%	72.4%
U_{1-2}	35695 A	25292 A
U_{1-3}	37869 A	24551 A
U_{2-3}	36887 A	16923 A
U_T	110451 A	66766 A

$$K(\mathbf{x}_i, \mathbf{x}_j) = e^{-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\sigma^2}}, \quad (1)$$

where σ^2 is the variance of the Gaussian kernel.

We investigate the performance of two different SVMs, one using the unsorted loads as training data and one using the sorted loads. To evaluate the performance of the different SVMs, the current unbalance between the different phases (between phases one and two (U_{1-2}), one and three (U_{1-3}), and two and three (U_{2-3})) and the total unbalance (sum of the unbalance between the different phases (U_T)) will give an indication which method performs the best. In total there were 1663 sets of loads used as training data and 500 sets of loads were used to test the load balancing. The results of the test data for each of the methods are shown in Table 1. Note that the current unbalance values shown, are the total over all 500 sets of loads in the test data set. From the table we can clearly see that the SVM that was trained with the sorted load data outperformed the SVM that was trained with the unsorted data. For 71.2% (356 loads out of 500) of the loads the sorted SVM gave the best load balancing, while the unsorted SVM gave the best balancing only for 28.8% (144 loads) of the loads. Looking at the unbalance factor, one can also see that for the sorted SVM the total unbalance over the 500 loads is 64.6% of the unbalance of the unsorted SVM. Thus, by sorting the loads, we could reduce the total unbalance by a significant percentage. We also compared the method with a heuristic method (which does not require prior training) for load balancing [8], where the results are shown in Table 2. We see that the SVM method with sorting outperforms both the SVM with unsorted loads and the heuristic method.

4 Conclusion

Phase and load balancing are important components to network and feeder re-configuration. In distribution automation these problems have to be continuously solved simultaneously to guarantee optimal performance of a distribution network. In this paper the phase balancing problem at the distribution transformers has been formulated as a current balancing optimization problem using SVM models. Before the load data is applied to the SVM, it is normalised and sorted in ascending order. It is shown that the SVM that was trained with the sorted load data has a much better load balancing than the SVM that was trained with the unsorted load data.

References

1. Chen, T., Cherg, J.: Optimal phase arrangement of distribution transformers connected to a primary feeder for system unbalance improvement and loss reduction using generic algorithm. *IEEE Trans. Power Systems* 17 (2000)
2. Civanlar, S., Grainger, J.: Distribution feeder reconfiguration for loss reduction. *IEEE Trans. PWRD-3*, 1227–1223 (1998)
3. Baran, M., Wu, F.: Network reconfiguration in distribution systems for loss reduction and load balancing. *IEEE Trans. Power Delivery* 7 (1989)
4. Ukil, A., Siti, W., Jordaan, J.: Feeder load balancing using neural network. In: Wang, J., Yi, Z., Żurada, J.M., Lu, B.-L., Yin, H. (eds.) *ISNN 2006. LNCS*, vol. 3972, pp. 1311–1316. Springer, Heidelberg (2006)
5. von Jouanne, A.: Assessment of voltage unbalance. *IEEE Trans. On Power Systems* 15(3) (2000)
6. Mathworks: *MATLAB Documentation - Neural Network Toolbox*. Version 6.5.0.180913a Release 13 edn. Mathworks Inc., Natick, MA (2002)
7. Pelckmans, K., Suykens, J., Van Gestel, T., De Brabanter, J., Lukas, L., Hamers, B., De Moor, B., Vandewalle, J.: *LS-SVMLab Toolbox User's Guide*, Version 1.5. Catholic University Leuven, Belgium (2003), <http://www.esat.kuleuven.ac.be/sista/lssvmlab/>
8. Ukil, A., Siti, M., Jordaan, J.: Feeder Load Balancing Using Combinatorial Optimization-based Heuristic Method. In: *13th IEEE International Conference on Harmonics and Quality of Power (ICHQP)*, Wollongong, New South Wales, Australia (2008)

Extracting Auto-Correlation Feature for License Plate Detection Based on AdaBoost

Haichun Tan¹, Yafeng Deng², and Hao Chen¹

¹ Department of Transportation Engineering, Beijing Institute of Technology,
Beijing, 100084, China
tanhc@bit.edu.cn

² Vimicro Corp. Beijing 100083, China
dengyafeng@gmail.com

Abstract. In this paper, a new method for license plate detection based on AdaBoost is proposed. In the proposed method, auto-correlation feature, which is ignored by previous learning-based method, is introduced to feature pool. Since that there are two types of Chinese license plate, one type is deeper-background-lighter-character and the other is lighter-background-deeper-character, training a detector cannot convergent. To avoid this problem, two detectors are designed in the proposed method. Experimental results show the superiority of proposed method.

Keywords: AdaBoost, License Plate Detection, Auto-Correlation.

1 Introduction

License Plate Recognition (LPR) system plays a very important role in intelligent traffic control and management. It has many applications recent years, for example, car park entrance and exit management, high way surveillance, and electric toll collection [1]. Among the three steps of LPR - license plate detection (LPD), character segmentation and character recognition, LPD is the most important since only the plate can be extracted correctly the following steps can perform well.

The most common approaches for license plate detection include texture [1] [2], edge characteristic [3], plate color [4], and learning-based approach [5]. Color is very useful when the lighting condition is good, but color information is not stable when the illumination changed. Texture/edge based methods are widely used for the advantage of finding plate candidates under different lighting conditions efficient and fast. These methods use the fact that there are characters in the plate, so the area contains rich edge and texture information.

In these methods, learning-based method is very efficient in many scenes. Different license plate features are learned from samples to extract license plate. Among learning-based methods, AdaBoost-based methods obtain good results. Luoka et al. [6] selected weak classifiers using AdaBoost from 6 types of features, which are x and y derivative and variance, for license plate detection. Xu et al. [7] used ten kinds of rectangle features selected by the AdaBoost algorithm to locate the regions of license plates. Zhang et al. [5] proposed a learning-based method using AdaBoost for license

plate detection under complex environments. They used both global statistical features, which are gradient density of the candidate and density variance of 12 sub-blocks, and local Haar-like features of gradient to detect the license plate. Arth et al. [8] proposed three steps detection: the detecting, the tracking and the post-processing step, in which the detector is based on the framework of Real-AdaBoost and the feature-pool is based on edge orientation histograms as proposed by Levi and Weiss [9]. However, for learning based methods, they always suffered from deferent types of license plate, poor illumination, complex background, noise etc. More important, for Chinese license plate detection, due to its specialization, for example, there are deeper-background-lighter-character and lighter-background-deeper-character license plates, using a uniform detector perform unsatisfied.

In this paper, a novel license plate detection algorithm is proposed based on AdaBoost. In the new method, the auto-correlation feature [10], which is a powerful feature for verifying license plate, is introduced to feature pool. To detect Chinese license plate, two detectors, one for deeper-background-lighter-character and one for lighter-background-deeper-character license plates, are trained respectively in the proposed method. Experimental results show the superiority of proposed method.

The rest of the paper is organized as follows. The features for AdaBoost, including auto-correlation feature, are defined in Section 2. The approach of selecting features are described in Section 3. Experimental results are shown in Section 4. Finally, a conclusion is summarized in Section 5.

2 License Plate Features

In previous approaches for license plate detection, texture [1] [2], edge characteristic [3], plate color [4], etc. are used. In these features, texture/edge based features are widely used for the advantage of finding plate candidates under different lighting conditions efficient and fast. These features use the fact that there are several characters in the plate, so the area contains rich edge and texture information. The methods based on AdaBoost mentioned in section 1 almost use the texture/edge features of license plate. In the proposed method, the texture/edge features are also emphasized. But different from these methods, auto-correlation features ignored by above mentioned methods, which is also a powerful feature for license plate detection, is considered.

2.1 Auto-Correlation Feature

The auto-correlation can be used to distinguish the image which has several interval blocks. Sin et al. [11] propose a powerful method based auto-correlation to detect text in scene images. In his method, the auto-correlation of the Fourier spectrum is calculated. The peaks are aligned along the slopes at an equal distance in text area, while non-text blocks do not possess such a sequence of peaks. Chen et al. [10] propose a license plate verification method using auto-correlation and projection based binary image. These methods show that the auto-correlation is a promise feature for license plate detection. However, the method is based on some rules to verify the license plates. For a rule-based method, setting effective rules for license plate detection under various environments is difficult. In the proposed method, auto-correlation feature is introduced to detect the license plate, and AdaBoost algorithm is adopted for automatically selecting auto-correlation features.

Fig. 1 shows an example of calculating the auto-correlation feature of a license plate. Fig. 1(a) shows the original license plate image. Then, the original image is binarized using Otsus method [12]. The result is shown in Fig. 1(b). To make the characteristic of several interval blocks more clearly, a $R*1$ rank-filter is employed by scanning the image in top-down direction from left to right, where R is set 5 in our algorithm. The binary image after rank filter process is shown in Fig. 1(c). To speed the calculation of auto-correlation feature, the horizontal projection of the binary image, as shown in Fig. 1(d) is processed and normalized.

The horizontal projection is normalized by subtracting the mean value. Then the auto-correlation feature can be obtained. Let B denote the normalized horizontal projection, w represents the width of B , x is the index of B . The auto-correlation curve S , as shown in Fig. 1(e) is defined by the following equation:

$$S(n) = \sum_{x=1}^w B(x+n) \bullet B(x) \quad (1)$$

To speed the calculation of auto-correlation and capture the frequency feature of several interval characters, 1-D Fast Fourier Transform (FFT) defined in equation (2) is applied to obtain the Fourier spectrum of auto-correlation.

$$X_k = \sum_{n=1}^N S(n) \exp\left(-\frac{j2\pi kn}{N}\right) \quad k=0, 1, \dots, N \quad (2)$$

The example of Fourier spectrum of auto-correlation curve is shown in Fig. 1(f). In the proposed method, the Fourier spectrum of auto-correlation curve is selected by AdaBoost algorithm and used as license plate feature to detect license plate.

3 Feature Selection

To select and use the important features, the AdaBoost algorithm proposed by Freund et al. [13] is adopted. The essence of the method is to build a strong classifier by boosting a group of weak classifier. Viola et al. [14] use this method to detect face objects and made great success in detection rate and in real-time usage.

The framework of our method is shown in Fig. 2. In the method, the detector is based upon a cascade classifier [14]. Comparing with the method in previous methods [5], the proposed method has two main different points. Firstly, the feature pool includes only auto-correlation feature. The main purpose of only using auto-correlation feature is to show the promising capability of detecting license plate. Secondly, two detectors instead of one detector in previous methods are used to detect license plate. According to the color combination of Chinese license plate, there are five different types of plates in China (denote by character color-plate color): blue-white, black-yellow, white-black, red-white, and black-white. After binarization, there are two types of license plate: deeper-background-lighter-character and lighter-background-deeper-character license plates. If only one detector is used, the training process can not converge properly. Then, in the proposed method, two detectors are trained, one for deeper-background-lighter-character and one for lighter-background-deeper-character license plates, respectively. The following briefly introduces the framework of proposed method.

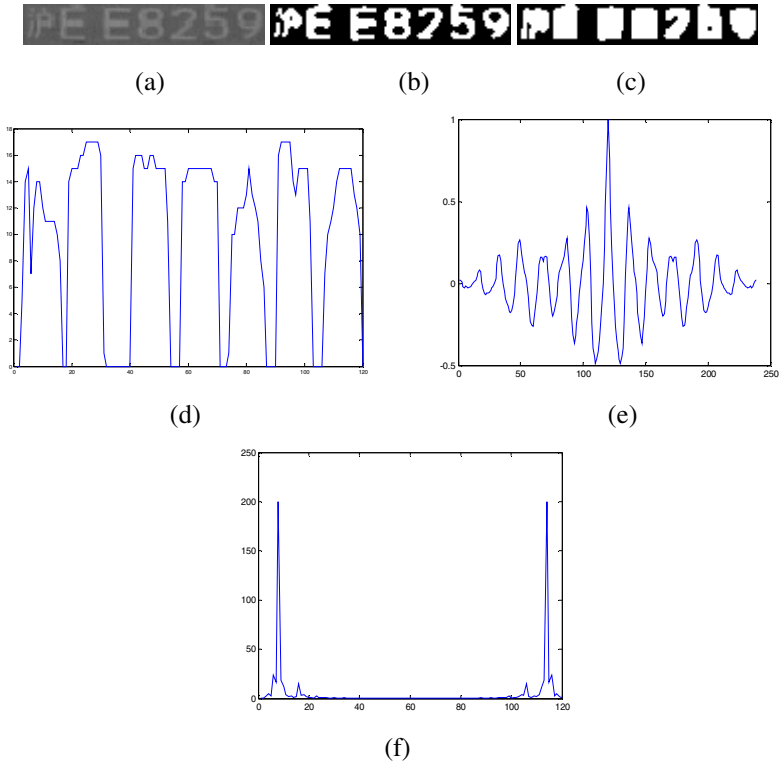


Fig. 1. An example of calculating auto-correlation feature. (a) Original license plate; (b) Binary image; (c) Binary image after rank filter; (d) Horizontal projection of binary image; (e) auto-correlation curve; (f) Fourier spectrum of auto-correlation curve.

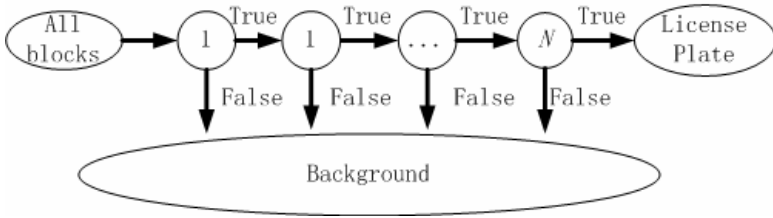


Fig. 2. The working flow of our method using cascade classifier

3.1 Training

According to the above analysis, the training method of AdaBoost is employed. The weak classifier $h_j(x)$ consists of a feature f_j , a threshold θ_j and a parity p_j indicating the direction of the inequality sign:

$$h_j(x) = \begin{cases} 1 & \text{if } p_j f_j < p_j \theta_j \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

Here x is a 48×16 pixel sub-window of an input image. The learning processing is described in following.

- Given example image $(x_1, y_1), \dots, (x_n, y_n)$ where $y_i = 0, 1$ for negative and positive examples respectively.
- Initialize weights $w_{1,i} = \frac{1}{2m}, \frac{1}{2l}$ for $y_i = 0, 1$ respectively, where m and l are the number of negatives and positives respectively.
- For $t=1, \dots, T$:

(1). Normalize the weights,

$$w_{t,i} \leftarrow \frac{w_{t,i}}{\sum_{j=1}^n w_{t,j}} \quad (4)$$

So that w_t is a probability distribution.

(2) For each feature, j , train a classifier h_j which is restricted to using a single feature. The error is evaluated with respect to

$$\varepsilon_j = \sum_i w_i |h_j(x_i) - y_i| \quad (5)$$

(3) Choose the classifier, h_t , with the lowest error ε_t .

(4) Update the weights:

$$w_{t+1,i} = w_{t,i} \beta_i^{1-e_i} \quad (6)$$

where $e_i = 0$ if example x_i is classified correctly, $e_i = 1$ otherwise, and $\beta_i = \frac{\varepsilon_t}{1 - \varepsilon_t}$

- The final strong classifier is:

$$h_j(x) = \begin{cases} 1 & \sum_{t=1}^T \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^T \alpha_t \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

3.2 Testing

After the classifier is built, a search window of 96×16 is scanning across the image, several matches are found near license plate. To handle the case of multiple matches, the method proposed in [6] is used to yields a single location that is closest to the true location of the license plate.

The detection is also implemented in multiple scales. In order to detect license plate of variant sizes, the search window is scale up from 96×16 to 240×40 , with a scaling factor of 1.2.

4 Experimental Results

Our database consists of 704×288 JPEG formatted color images. These images are taken from a city road as a part of intelligent traffic surveillance system. In order to

analyze the performance of proposed approach, 562 images from a day surveillance are tested. Some examples of the images are shown in Fig. 3. Various Chinese license plates are captured in day and night illumination conditions. Among these positive images, 412 images containing license plate, in which 250 images are deeper-background-light-character type, are used for training. All training positive samples are labeled and cut by hand and normalized to 96x16 pixels. 150 license plates including 100 deeper-background-light-character license plates are used for test. 6747 negative samples are extracted by randomly selecting from 300 images without license plate. The negative samples used in AdaBoost learning procedure are the false positive samples obtained from the previous layers of the cascade classified. Two seven-layer cascade classifiers are trained for license plate detection. One detector is for deeper-background light-character license plate, and other is for light-background deeper-character case.



Fig. 3. Some examples of the license plate in database

To test the efficiency of the proposed method, the method proposed in [5] is also realized and tested. In the experiments, 149 license plates are detected correctly while only 1 false positive detected by the proposed method. This is due to the powerful feature of auto-correlation is used in the proposed method. For the method in [5], 129 license plates are found out while 6 false positive detected. The ROC curve is given in Fig. 4, where the Haar-like method represents the method proposed in [5]. From the ROC curve, it can be found that the proposed method obtains higher detection rate while lower false positive rate comparing with the method proposed in [5].

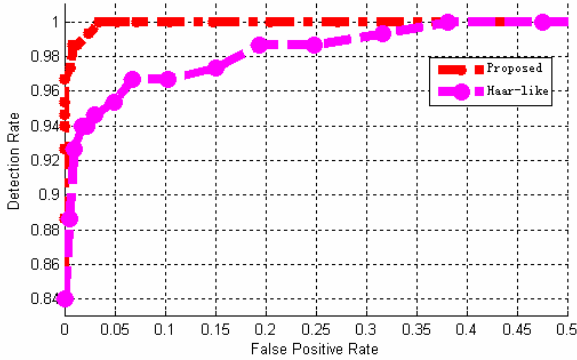


Fig. 4. The ROC curve of proposed method and the method in [5]

5 Conclusion

In this paper, a new method based on AdaBoost is proposed. There are two contributions in the new method. Firstly, auto-correlation feature, which is powerful feature for detecting license plate character, are introduced to feature pool. Secondly, considering the characteristic of Chinese license plate that there are two types of license plate: deeper background-lighter character and lighter background-deeper character license plates, two detectors are designed to extract different license plates respectively. Experimental results in a day city road surveillance system show the efficiency of the proposed method. 412 positive images and 7647 negative images are used for training. 150 license plates are used for testing. The result of 149 license plates are detected correctly while 1 false positive is obtained by the proposed method, comparing with 86% detection rate with 6 false positive of the method in [5]. The ROC curve shows the efficiency of auto-correlation feature for license plate detection.

In the proposed method, only the Fourier spectrum of auto-correlation is selected by AdaBoost. However, many other features can describe the license plate powerfully. If other features are considered, the performance of license plate detection may be improved. Since the auto-correlation feature uses convolution and is based on binary images, how to get the auto-correlation feature faster like the process of obtaining Haar-like feature is challenged. These are both our future works.

Acknowledgements. This work is supported by EYSCR fund of BIT (2007Y0305), Beijing Training Programming Foundation for the Talents (20081D1600300343), and State 863 projects (NO.2006AA11Z213).

References

- [1] Shapiro, V., Gluhchev, G., Dimov, D.: Towards a multinational car license plate recognition system. *Machine Vision Application* 17(3), 173–183 (2006)
- [2] Anagnostopoulos, C.N.E., Anagnostopoulos, I.E.: A License Plate-Recognition Algorithm for Intelligent Transportation System Applications. *IEEE Transactions on Intelligent Transportation System* 7(3) (2006)

- [3] Zheng, D., Zhao, Y., Wang, J.: An efficient method of license plate location. *Pattern Recognition Letters*, 2431–2438 (2005)
- [4] Chang, S., Chen, L., Chung, Y., Chen, S.: Automatic License Plate Recognition. *IEEE Transactions on Intelligent Transportation System* 5(3) (2004)
- [5] Zhang, H., Jia, W., He, X., Wu, Q.: Learning-Based License Plate Detection Using Global and Local Features. In: *ICPR*, vol. 2, pp. 1102–1105 (2006)
- [6] Dlagnekov, L., Belongie, S.: Recognizing cars. Technical Report CS2005-0833, UCSD University of California, San Diego (2005)
- [7] Xu, X., et al.: A Method of Multi-view Vehicle License Plates Location Based on Rectangle Features. In: *ICSP*, p. 4129276 (2006)
- [8] Arth, C., Limberger, F., Bischof, H.: Real-time License Plate Recognition on An Embedded DSP-Platform. In: *CVPR*, pp. 791–798 (2007)
- [9] Levi, K., Weiss, Y.: Learning object detection from a small number of examples: the importance of good features. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, pp. 53–60 (2004)
- [10] Chen, H., Tan, H., Wang, J.: Correlation Based Method of Candidate Verification for License Plate Extraction. *World Congress on ITS* (2007)
- [11] Sin, B., Kim, S., Cho, B.: Locating characters in scene images using frequency features. In: *ICPR*, vol. 3, pp. 489–492 (2002)
- [12] Otsu, N.: A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man, and Cybernetics* 9, 62–66 (1979)
- [13] Freund, Y.: An Adaptive Version of the Boost by Majority Algorithm. *Machine Learning* 43(3), 293–318 (2001)
- [14] Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: *CVPR*, pp. 511–518 (2001)

Evolutionary Optimization of Union-Based Rule-Antecedent Fuzzy Neural Networks and Its Applications

Chang-Wook Han

Department of Electrical Engineering, Dong-Eui University,
995 Eomgwangno, Busanjin-gu, Busan, 614-714, South Korea
cwhan@deu.ac.kr

Abstract. A union-based rule-antecedent fuzzy neural networks (URFNN), which can guarantee a parsimonious knowledge base with reduced number of rules, is proposed. The URFNN allows union operation of input fuzzy sets in the antecedents to cover bigger input domain compared with the complete structure rule which consists of AND combination of all input variables in its premise. To construct the URFNN, we consider the union-based logic processor (ULP) which consists of OR and AND fuzzy neurons. The fuzzy neurons exhibit learning abilities as they come with a collection of adjustable connection weights. In the development stage, genetic algorithm (GA) constructs a Boolean skeleton of URFNN, while gradient-based learning refines the binary connections of GA-optimized URFNN for further improvement of the performance index. A cart-pole system is considered to verify the effectiveness of the proposed method.

1 Introduction

Fuzzy logic, proposed by Zadeh in 1965, is a logic with fuzzy truth, fuzzy connectives and fuzzy rules of inference rather than the conventional two-valued or even multi-valued logic [1]. It combines multi-valued logic, probability theory and a knowledge base to mimic human thinking by incorporating the uncertainty inherent in all physical systems. Relying on the human nature of fuzzy logic, an increasing number of successful applications have been developed, like automatic process control [2], pattern recognition systems [3] and so forth.

A common practice in traditional approaches to building fuzzy rule bases is to use all AND-combinations of input fuzzy sets as rule antecedents [4][5]. In this case, the number of such combinations increases exponentially with the input number [6]. To reduce the size of the rule bases, an union-based rule-antecedent fuzzy neural networks (URFNN) is proposed in this paper. The URFNN allows union operation of input fuzzy sets in the antecedents to cover bigger input domain compared with the complete structure which consists of AND combinations of fuzzy sets of all input variables in its premise. Basically, the URFNN is constructed with the aid of union-based logic processor (ULP) which consists of OR and AND fuzzy neurons. The fuzzy neurons exhibit learning abilities as they come with a collection of adjustable connection weights [6][7]. This paper aims at constructing a binary structure of

URFNN by genetic algorithm (GA) [8], and subsequently further refining the binary connections by gradient-based learning introduced in [6][7]. The effectiveness of the URFNN is demonstrated by stabilizing an inverted pendulum system.

2 Structure of the URFNN

Before discussing the architecture of the URFNN, we will briefly remind AND and OR fuzzy neurons and then move on to the ULP which is the basic logic processing unit of the URFNN.

2.1 OR and AND Fuzzy Neurons

As originally introduced in [6][7], fuzzy neurons emerge as result of a vivid synergy between fuzzy set constructs and neural networks. In essence, these neurons are functional units that retain logic aspects of processing and learning capabilities characteristic for artificial neurons and neural networks. Two generic types of fuzzy neurons are considered:

AND neuron is a nonlinear logic processing element with n -inputs $x \in [0, 1]^n$ producing an output y governed by the expression

$$y = \text{AND}(x; w) \quad (1)$$

where w denotes an n -dimensional vector of adjustable connections (weights). The composition of x and w is realized by an t - s composition operator based on t - and s -norms, that is

$$y = \prod_{i=1}^n (w_i \text{ s } x_i) \quad (2)$$

with “ s ” denoting some s -norm and “ t ” standing for a t -norm. As t -norms (s -norms) carry a transparent logic interpretation, we can look at as a two-phase aggregation process: first individual inputs (coordinates of x) are combined or-wise with the corresponding weights and these results produced at the level of the individual aggregation are aggregated and-wise with the aid of the t -norm.

By reverting the order of the t - and s -norms in the aggregation of the inputs, we end up with a category of OR neurons,

$$y = \text{OR}(x; w) \quad (3)$$

that is

$$y = \sum_{i=1}^n (w_i \text{ t } x_i) \quad (4)$$

We note that this neuron carries out some and-wise aggregation of the inputs followed by the global or-wise combination of these partial results.

Some obvious observations hold:

- (i). For binary inputs and connections, the neurons transform to standard OR and AND gates.

- (ii). The higher the values of the connections in the OR neuron, the more essential the corresponding inputs. This observation helps eliminate irrelevant inputs; the inputs associated with the connections whose values are below a certain threshold are eliminated. An opposite relationship holds for the AND neuron; here the connections close to zero identify the relevant inputs.
- (iii). The change in the values of the connections of the neuron is essential to the development of the learning capabilities of a network formed by such neurons; this parametric flexibility is an important feature to be exploited in the design of the networks.

These two types of fuzzy neurons are fundamental building blocks used in the design of logic expressions supporting the development of logic-driven models.

2.2 The URFNN and Its Two-Step Optimizations

To construct the URFNN, we first elaborate on the ULP which consists of OR and AND fuzzy neurons, as shown in Fig. 1, where, $\mu_N(x_i)$, $\mu_Z(x_i)$ and $\mu_P(x_i)$ are the membership grades of the fuzzy sets N (negative), Z (zero) and P (positive) for the input variable x_i , $i=1,2,3,4$, respectively. The OR and AND fuzzy neurons realize pure logic operations on the membership values and exhibit learning abilities as being introduced in [6][7]. In this paper, we consider these triangular norms and co-norms to be a product operation and probabilistic sum, respectively.

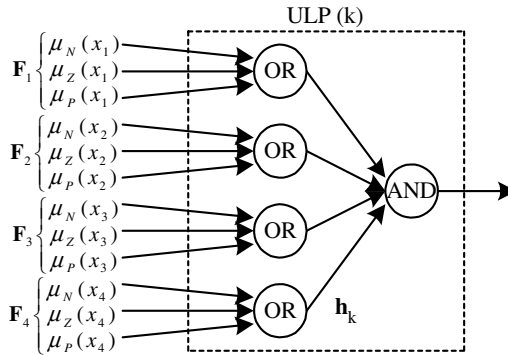


Fig. 1. Structure of an ULP

An important characteristic of ULP is that union operation of input fuzzy sets is allowed to appear in their antecedents, i.e., incomplete structure. For fuzzy system of complex processes with high input dimension, the ULP is preferable because it achieves bigger coverage of input domain compared with the complete structure. For example, consider a system with x_1, x_2 as its inputs and y as its output characterized by three linguistic terms, N, Z and P, respectively. The incomplete structure rule ‘If $x_1=N$ then $y=N$ ’ covers the following three complete structure rules:

- (i). If $(x_1=N)$ and $(x_2=N)$ then $y=N$
- (ii). ‘If $(x_1=N)$ and $(x_2=Z)$ then $y=N$
- (iii). ‘If $(x_1=N)$ and $(x_2=P)$ then $y=N$

Similarly, the rule ‘If $(x_1=N$ or $Z)$ and $(x_2=N$ or $Z)$ then $y=N$ ’ covers the following four complete structure rules:

- (i). If $(x_1=N)$ and $(x_2=N)$ then $y=N$
- (ii). If $(x_1=N)$ and $(x_2=Z)$ then $y=N$
- (iii). If $(x_1=Z)$ and $(x_2=N)$ then $y=N$
- (iv). If $(x_1=Z)$ and $(x_2=Z)$ then $y=N$

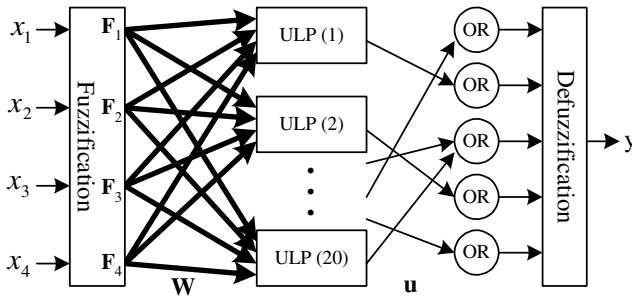


Fig. 2. Structure of an URFNN with 4 input and 1 output variables characterized by 3 and 5 fuzzy sets, respectively (NU=20)

Fig. 2 describes the URFNN constructed with the aid of ULPs. The OR neurons in the output layer are placed to aggregate the outputs of ULPs for each corresponding consequences. In Fig. 2, the connections to the ULPs are described as bold lines which contain a set of connection lines, refer to Fig. 1. The only parameter that has to be controlled in the URFNN is the number of ULP (NU), which will be set large enough in the experiment. A conflict occurs in the rule bases if there exist two rules which have overlapping AND combinations but different linguistic consequences. Let us consider the following two rules to count the number of conflict (NC):

- (i). If $(x_1=N$ or $Z)$ and $(x_2=N$ or $Z)$ then $y=N$
- (ii). If $(x_1=Z$ or $P)$ and $(x_2=N$ or $Z)$ then $y=Z$

In this case, NC=2 because the antecedents ‘ $(x_1=Z)$ and $(x_2=N)$ ’ and ‘ $(x_1=Z)$ and $(x_2=Z)$ ’ have different consequences. Therefore, NC should be checked for all possible pairs of different consequences of the rule bases. It will be included in evaluating the performance index to remove the conflict from the rule bases later on.

For the development of the URFNN, GA attempts to construct a Boolean structure of URFNN by selecting the most essential binary connections that shape up the architecture. GA optimizes the binary connections \mathbf{W} and \mathbf{u} , as shown in Fig. 2. All the connections to AND neurons, \mathbf{h}_k , in the ULPs are initialized as zero (valid connection). It is worth noting that if the connection weights from \mathbf{F}_1 to an OR neuron in the ULP are all-one, that is, x_1 is ‘don’t care’ in this ULP, the corresponding connection to

AND neuron should be modified as one (invalid connection). It is obvious that the following two cases lead to an invalid rule antecedent:

- (i). All-one connection weights to an ULP, meaning that all input variables are ignored in the antecedent.
- (ii). All-zero connection weights to an OR neuron in the ULP, i.e., empty fuzzy set for the corresponding input variable.

Therefore, these ULPs will be removed from the rule bases and finally the compact rule bases will be established. To avoid the rule confliction, the output of an ULP should be connected to only one of the OR neurons in the output layer. Since the GA is a very popular optimization algorithm considered in many application areas, we do not elaborate on this. For more details about the GA, please refer to [8].

Once a Boolean skeleton has been constructed by GA, we concentrate on the detailed optimization of the valid binary connections with the aid of gradient-based learning [6][7]. It is apparent that the number of valid binary connections is much smaller than the number of all possible connections in the URFNN. Therefore, the gradient-based learning that has a local search ability is considered to refine the reduced number of valid binary connections optimized by GA. The gradient-based learning refinement involves transforming binary connections into the weights in the unit interval. This enhancement aims at further improvement in the value of the performance index. Obviously we do not claim that the gradient-based learning is the most effective learning method for this purpose. We intend to show how much the connection refinement affects the performance of the URFNN. For more details about gradient-based learning for AND and OR fuzzy neurons, please refer to [6][7].

3 Experimental Results

To show the effectiveness of the proposed URFNN, a cart-pole system, a well-known nonlinear system, was considered. The objective is to bring the cart to center with a vertical pole. The system has four state variables which are θ (angle of the pole with the vertical), $\dot{\theta}$ (angular velocity of the pole), x (position of the cart) and \dot{x} (linear velocity of the cart). The nonlinear differential equations for a cart-pole system are described as follows [9]:

$$\ddot{\theta} = \frac{(M+m)g \sin \theta - \cos \theta [f + mL\dot{\theta}^2 \sin \theta - \mu_c \operatorname{sgn}(\dot{x})] - \frac{\mu_p(M+m)\dot{\theta}}{mL}}{\frac{4}{3}(M+m)L - mL \cos^2 \theta} \quad (5)$$

$$\ddot{x} = \frac{f + mL(\dot{\theta}^2 \sin \theta - \ddot{\theta} \cos \theta)}{M+m} - \mu_c \operatorname{sgn}(\dot{x})$$

The parameter values used in this simulation were set as the same as [9] except the failure conditions, where $|\theta| > 0.2 \text{ rad}$ or $|x| > 0.5 \text{ m}$. The following optimization parameters were considered: (i) GA parameters are population size=100; generation number=200; crossover rate=0.9; mutation rate=0.03; (ii) gradient-based learning parameters are learning rate=0.01; iteration number=500; (iii) others are time

step=0.01s; simulating number of time steps for each initial condition $q=500$; $NU=20$. For the GA, standard version including two-point crossover based on the boundary between binary (connections \mathbf{W} for antecedents) and integer (connections \mathbf{u} for consequences) codes was used. GA optimized 240 (12x20) binary parameters and 20 integer parameters that have the integer values of $\{1,2,\dots,5\}$. The parameters of gradient-based learning were set to perform fine learning of the binary connections. The following normalized performance index (fitness function) that has to be maximized was used:

$$Q = \frac{1}{8} \sum_{cond1}^{cond8} \left\{ \frac{q_s}{q} \left[1 - \frac{1}{2q} \sum_{j=1}^q \left(\frac{|\theta_j|}{\theta_{fail}} + \frac{|x_j|}{x_{fail}} \right) \right] \right\} \left\{ 1 - \frac{NC}{NC_{max}} \right\} \quad (6)$$

where θ_{fail} is 0.2rad, x_{fail} is 0.5m, q_s is the survival time steps for each trial, and NC_{max} is the maximum NC which is set as 10. If NC exceeds NC_{max} , $NC=NC_{max}$. For the evaluation of the performance index, eight different sets of initial conditions (cond1~cond8) were considered to cover wide range of input spaces. Because our focal point is URFNN and its two-step optimization, we assume that fuzzy sets of the input/output variables are given in advance as 3/5-uniformly distributed triangular membership functions with an overlap of 0.5 and left unchanged. For the defuzzification, center of area method was used. Ten independent simulations for the optimization of URFNN have been performed, and the results are shown in Table 1. This table describes the average best performance index after GA and gradient-based learning over ten independent simulations as well as the maximum, minimum and average number of valid ULP (incomplete structure rules) for ten optimized URFNN. As can be seen, the optimized URFNN has at most 15 incomplete structure rules covering most of the essential input domain without linguistic conflict, and besides, the gradient-based learning further refines the valid binary connections optimized by GA.

Table 1. Results of the optimized URFNN

After GA	After gradient-based learning	Max rule	Min rule	Average rule
0.912	0.951	15	13	14.1

Fig. 3 illustrates the results of testing simulations for the optimized URFNN with a set of initial condition $(\theta, \dot{\theta}, x, \dot{x})=(0, 0, -0.25, 0)$ which is independent of the sets for the optimization. To compare the results, the conventional fuzzy controller (CFC) which has all possible AND combinations (81 complete structure rules) as rule antecedents was considered.

Obviously the performance of CFC is better than that of URFNN, but the control results indicate that the reduced rule bases of URFNN are enough to center the cart with a vertical pole, and moreover, the performance of URFNN outperforms that of [9].

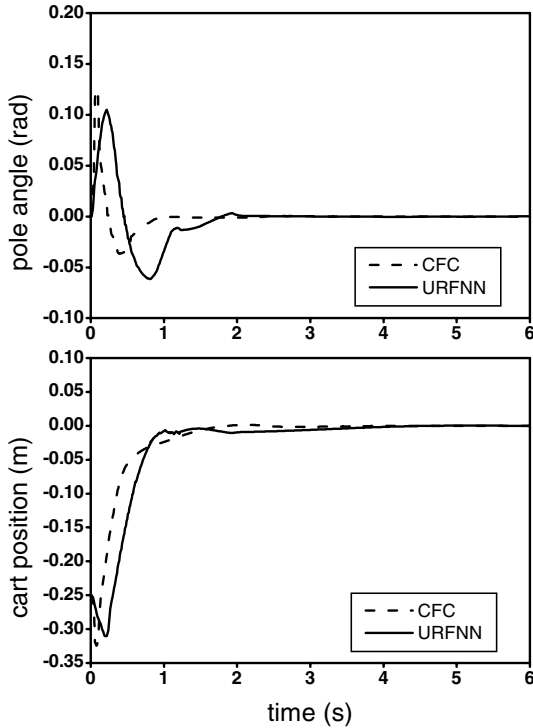


Fig. 3. Control results using a testing initial condition

4 Conclusions

The URFNN that allows union operation to appear in their antecedents for the parsimonious rule bases has been demonstrated. It has been constructed with the aid of fuzzy neurons and two-step optimizations, where GA develops a Boolean skeleton and subsequently gradient-based learning further refines the binary connections. As can be seen from the simulation results, the incomplete structure of rule allows the URFNN to utilize only fewer rules without performance degradation.

References

1. Zadeh, L.A.: Fuzzy sets. *Inform. Control* 8, 338–353 (1965)
2. King, P.J., Mamdani, E.H.: The application of fuzzy control systems to industrial processes. *Automatica* 13(3), 235–242 (1977)
3. Pal, S.K., King, R.A., Hashim, A.A.: Image description and primitive extraction using fuzzy set. *IEEE Trans. Syst. Man and Cybern. SMC-13*, 94–100 (1983)
4. Karr, C.L.: Design of an adaptive fuzzy logic controller using a genetic algorithm. In: *Proc. Int. Conf. on Genetic Algorithms*, pp. 450–457 (1991)

5. Homaifar, A., McCormick, E.: Simultaneous design of membership functions and rule sets for fuzzy controllers using genetic algorithms. *IEEE Trans. Fuzzy Systems* 3(2), 129–139 (1995)
6. Pedrycz, W., Reformat, M., Han, C.W.: Cascade architectures of fuzzy neural networks. *Fuzzy Optimization and Decision Making* 3(1), 5–37 (2004)
7. Pedrycz, W.: Fuzzy Neural Networks and Neurocomputations. *Fuzzy Sets and Systems* 56, 1–28 (1993)
8. Goldberg, D.E.: *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, Reading (1989)
9. Liu, B.D., Chen, C.Y., Tsao, J.Y.: Design of adaptive fuzzy logic controller based on linguistic-hedge concepts and genetic algorithms. *IEEE Trans. Syst. Man and Cybern.-B* 31(1), 32–53 (2001)

Improving AdaBoost Based Face Detection Using Face-Color Preferable Selective Attention

Bumhwi Kim¹, Sang-Woo Ban², and Minho Lee¹

¹School of Electrical Engineering and Computer Science, Kyungpook National Univ.,
1370 Sankyuk-Dong, Puk-Gu, Taegu 702-701, Korea

²Dept. of Information and Communication Engineering, Dongguk Univ.,
707 Seokjang-Dong, Gyeongju, Gyeongbuk 780-714, Korea
{bhkim,mholee}@ee.knu.ac.kr, swban@dongguk.ac.kr

Abstract. In this paper, we propose a new face detection model, which is developed by combining the conventional AdaBoost algorithm for human face detection with a biologically motivated face-color preferable selective attention. The biologically motivated face-color preferable selective attention model localizes face candidate regions in a natural scene, and then the Adaboost based face detection process only works for those localized face candidate areas to check whether the areas contain a human face. The proposed model not only improves the face detection performance by avoiding miss-localization of faces induced by complex background such as face-like non-face area, but can enhance a face detection speed by reducing region of interests through the face-color preferable selective attention model. The experimental results show that the proposed model shows plausible performance for localizing faces in real time.

Keywords: Face detection, selective attention, saliency map, AdaBoost.

1 Introduction

In the last five years, face and facial expression recognition have attracted much attention though they have been studied for more than 20 years by psychophysicists, neuroscientists, and engineers [1]. Numerous methods have been developed to localize or detect faces in a visual scene [1, 2]. M. Yang et al, [1] have reviewed and classified those face detection methods into four major categories such as the knowledge-based methods, the feature invariant approaches, the template matching methods, appearance-based methods. According to the survey, no specific method has yet shown comparable performance with a human being. Biologically inspired vision system may provide a critical clue to overcome the limitations of the current artificial vision system. Recently, biologically motivated approaches have been developed by L. Itti., T. Poggio, and C. Koch [3, 4, 5]. And, attention models were introduced for face detection [6, 7]. However, they have not shown plausible results for the face attention problem in complex scenes until now. Conventional face detection models based on an AdaBoost algorithm show good performance in a real time environment even if they are not perfectly working [2].

In this paper, we propose a real time face candidates localizer which is implemented by combining a biologically motivated selective attention model with a well-known AdaBoost algorithm for face detection.

The proposed selective attention model considers a face color filtered intensity, an R-G color opponent and edge information for reflecting human face preference. Thus, the proposed selective attention model generates a saliency map for an input scene, which pop-outs face candidate areas having the face-like low level features. The proposed face preferable saliency map (SM) model reduces region of interests in an input scene, which plays important role for decreasing face detection processing time. Moreover, in order to reject non-face areas and correctly localize face areas in the selected face candidate areas, we consider a well-known AdaBoost algorithm based on Haar-like form features. Haar-like form features are properly matching face form features naturally generated by inner shape of human face.

This paper is organized as follows; Section 2 describes the proposed face localization model using face-color preferable SM model and an AdaBoost algorithm. The experimental results will be followed in Section 3. Section 4 presents further works, conclusions and discussions.

2 Biologically Motivated Selective Attention Model and AdaBoost Algorithm for Localizing Human Face

When humans pay attention to a specific object, the prefrontal cortex generates a competitive bias signal, related with the target object, to the infero-temporal (IT) and the V4 area [8]. Then, the IT and the V4 area generate top-down bias signals such as target object dependant color and form information, and those are transmitted to the low-level feature extraction stage part such as the striate cortex area including the lateral geniculate nucleus (LGN) in order to construct a filter to select a preferential area satisfying the target object dependant features.

In the proposed model as shown in Figure 1, therefore, we simply consider a skin color preferable attention model for face color perception and Haar-like form features for face form perception, of which all processes work in real time. A biologically motivated selective attention model with face-color preference can decide face candidate areas in a complex input scene. For the selected face candidate regions, an AdaBoost algorithm using the Harr-like form feature is applied to selectively localize human faces not in all regions of the input scene but only in the face candidate areas obtained by the face color preferable selective attention model.

Thus, we propose a face candidate localizer based on the biologically motivated bottom-up SM model as shown in Fig. 1. The bottom-up SM model can preferably focus on face candidate areas by a simple face-specific color bias filter using face color filtered intensity, an R-G color opponent and edge information of the R-G color opponent feature. Then, the candidate regions are checked how much the localized areas match up trained face form features based on Haar-like features.

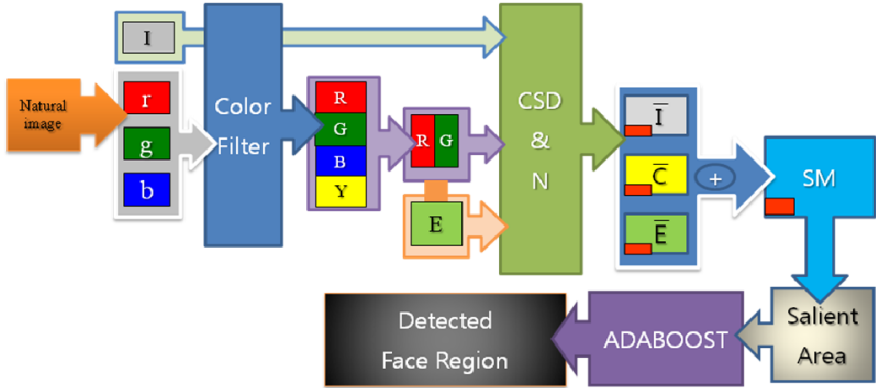


Fig. 1. The proposed face candidate localization model based on the saliency map (SM) model and the AdaBoost algorithm; r: red, g: green, b: blue, R: real red component, G: real green component, B: real blue component, Y: real yellow component, I: intensity, E: edge, RG: red-green opponent coding feature, CSD&N : center surround difference & normalization, \bar{I} : intensity feature map, \bar{E} : edge feature map, \bar{C} : color feature map, SM: saliency map

2.1 Face Color Preferable Selective Attention

In human visual processing, the intensity, edge and color features are extracted in the retina. These features are transmitted to the visual cortex through the LGN. While transmitting those features to the visual cortex, intensity, edge, and color feature maps are constructed using the on-set and off-surround mechanism of the LGN and the visual cortex. And those feature maps make a bottom-up SM model in the lateral intral-parietal cortex (LIP) [9].

In order to implement a human-like visual attention function, we consider the simplified bottom-up SM model proposed in [10]. The SM model reflects the functions of the retina cells, the LGN and the visual cortex. Since the retina cells can extract edge and intensity information as well as color opponency, we use these factors as the basic features of the SM model [10]. In order to provide the proposed model with face color preference property, the skin color filtered intensity feature is considered together with the original intensity feature. According to a given task to be conducted, those two intensity features are differently biased. For face preferable attention, a skin color filtered intensity feature works for a dominant feature in generating an intensity feature map. The ranges of red(r), green(g), blue(b) for skin color filtering are obtained from following rules in Eq. (1) [11].

$$\begin{aligned}
 &R > 95, G > 40, B > 20 \text{ and} \\
 &\max\{R, G, B\} - \min\{R, G, B\} > 15 \text{ and} \\
 &|R - G| > 15 \text{ and } R > G \text{ and } R > B
 \end{aligned} \tag{1}$$

And the real color components R, G, B, Y are extracted using normalized color coding [10]. According to our experiments, the real color component R among 4 real color components shows dominant contribution for face color plausible filtering. Moreover, RG color opponent coding features also show a discriminate characteristic

between face and non-face area. Instead, BY color opponent coding feature has a little contribution to discriminate whether face or non-face area. Therefore, in the proposed model, only the real color component R and RG color opponent feature are considered to generate a skin color filter, which also plays a role for reducing computation time as well as getting better skin color filtering performance.

Actually, considering the function of the LGN and the ganglion cells, we implement the on-center and off-surround operation by the Gaussian pyramid images with different scales from 0 to n -th level, whereby each level is made by the sub-sampling of 2^n , thus it is able to construct four feature bases such as the intensity (I), and the edge (E), and color (RG and BY) [9, 10]. This reflects the non-uniform distribution of the retina-topic structure. Then, the center-surround mechanism is implemented in the model as the difference operation between the fine and coarse scales of the Gaussian pyramid images [10].

Consequently, three feature maps are obtained by the following equations.

$$I(c, s) = |I(c) \ominus I(s)| \quad (2)$$

$$E(c, s) = |E(c) \ominus E(s)| \quad (3)$$

$$RG(c, s) = |R(c) - G(c)| \ominus |G(s) - R(s)| \quad (4)$$

where “ \ominus ” represents interpolation to the finer scale and point-by-point subtraction, c and s are indexes of the finer scale and the coarse scale, respectively. Totally, 18 features are computed because three features individually have 6 different scales [10, 11]. Features are combined into three feature maps as shown in Eq. (5) where \bar{I} , \bar{E} and \bar{C} stand for intensity, edge, and color feature maps, respectively. These are obtained through across-scale addition “ \oplus ” [10].

$$\begin{aligned} \bar{I} &= \bigoplus_{c=2}^3 \bigoplus_{s=c+2}^{c+3} N(I(c, s)), \\ \bar{E} &= \bigoplus_{c=2}^3 \bigoplus_{s=c+2}^{c+3} N(E(c, s)), \\ \bar{C} &= \bigoplus_{c=2}^3 \bigoplus_{s=c+2}^{c+3} N(RG(c, s)) \end{aligned} \quad (5)$$

Consequently, the three feature maps such as \bar{I} , \bar{E} and \bar{C} can be obtained by the center-surround difference and normalization (CSD&N) algorithm [10]. A SM is generated by the summation of these three feature maps as shown in Eq. (6).

$$SM = \bar{I} + \bar{E} + \bar{C} \quad (6)$$

The salient areas are obtained by selecting areas with relatively higher saliency in the SM. In order to decide salient area, the proposed model generates binary data for each selected face candidate area using Otsu’s threshold method in the SM [12]. Then, the proposed model makes a group of segmented areas using a labeling method for each binary face candidate area.

After obtaining the candidate salient areas for human face, the obtained face candidate areas are used as input of the AdaBoost algorithm.

2.2 Face Detection Using AdaBoost Algorithm

We adapted an AdaBoost approach using simple Haar-like features as the face detection algorithm for correctly localizing faces in the face candidate regions selectively selected by the face-color preferable SM model.

There are two data sets for face feature extraction and learning for the AdaBoost model. One is called a positive dataset in which every image has a face. The other for non-face images set is called a negative dataset. For two data sets, Haar-like features are extracted in order to select the proper features and train the AdaBoost face detection model. As well known, the AdaBoost learning algorithm is used to boost the classification performance of a simple learning algorithm [2]. The AdaBoost approach is working by combining a collection of weak classification functions to form a stronger classifier [2].

For each feature, the weak learner determines the optimal threshold classification function, such that a minimum number of examples are misclassified.

According to our experiments, a face detection model using AdaBoost algorithm based on Haar-like form features shows wrong face detection results in some cases, which can be avoided by considering face color preferable selective attention as a preprocessor. In addition, the AdaBoost algorithm takes longer processing time for some scenes with complex orientation features as well as it generates wrong face detection results for the scenes with complex background. In contrast, the proposed model can enhance the face detection performance by reducing the computation load as well as increasing face detection accuracy especially for a natural scene with complex orientation information.

3 Experimental Results

Figure 2 shows a simulation process of the proposed model. At first, the proposed model extracts intensity, RG opponent color features and edge from an input color scene. Skin color areas in intensity and RG opponent color feature images only are filtered by applying a previously obtained skin color filter, which works for reflecting face-color preferable property. Then, three feature maps are generated by a center-surround difference and normalization process for edge feature, skin color filtered intensity feature, and RG opponent color feature, respectively. Each feature map represents a degree of relative saliency of each area of an input scene just for the corresponding feature. Next, three different feature maps are integrated in order to generate a saliency map that represents a degree of relative saliency of each area of an input scene in all kinds of features point of view. The proposed model selectively decides salient areas with suitable size of an ROI in the obtained saliency map, which are assumed as candidate face areas and used as input of AdaBoost. Finally, AdaBoost decides whether each candidate face area is face area or not.

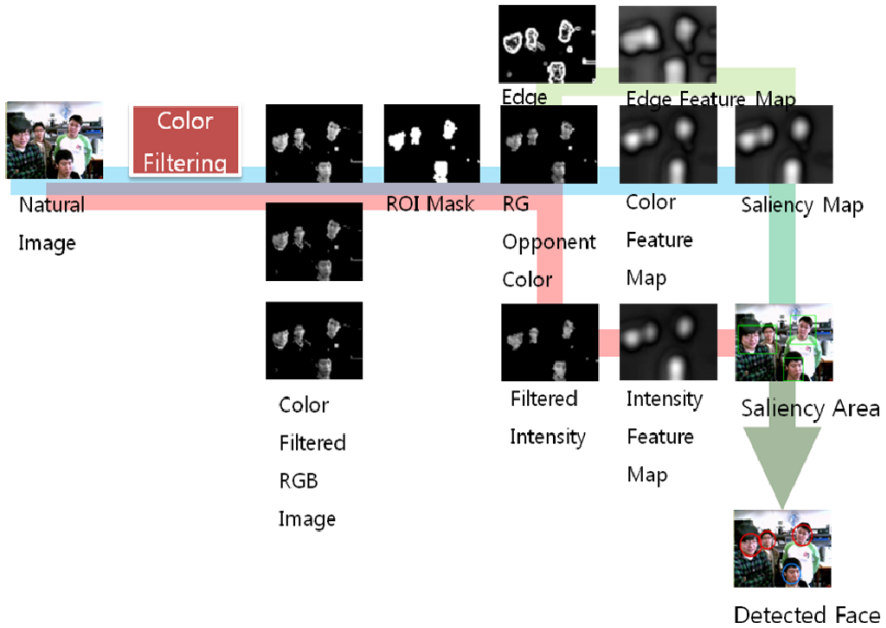


Fig. 2. The experimental result of face localization by the proposed model

As mentioned in Section 2.2, the face detection model using only the AdaBoost algorithm based on Haar-like form features generates some wrong face detection results. Figure 3 (a) shows an example with a wrong face detection case which caused by considering Haar-like form feature only in an intensity image by the AdaBoost algorithm. In this case a shirt is wrongly detected as a face since the intensity distribution in a shirt looks like a face. Those kinds of cases can be avoided by the proposed model as shown in Fig. 3 (b). A shirt is not selected as a face candidate area by the proposed face-color preferable attention model as shown in Fig. 3 (d). The candidate areas are decided based on the SM as shown in Fig. 3 (c), which is obtained from the face-color preferable attention model.

Also, as shown in Fig. 4 (a), the AdaBoost algorithm may detect faces in duplicate way for the same face area, which can be also avoided by the proposed model considering only properly decided attention area as shown in Figs. 4 (b) and (c).

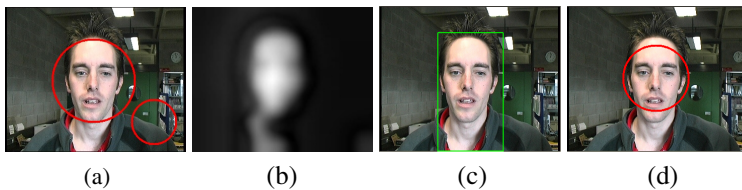


Fig. 3. Comparison of face detection between an AdaBoost algorithm based on Haar-like form features and the proposed method; (a) Face detection result by the AdaBoost algorithm, (b) Face color preferable SM, (c) ROI areas, (d) Face detection result by the proposed model

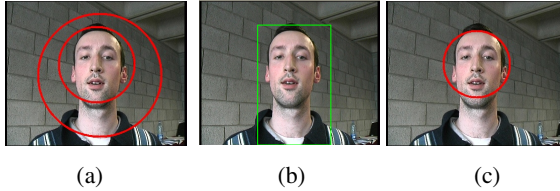


Fig. 4. Another comparison for face detection; (a) Duplicated face detection by an AdaBoost algorithm, (b) and (c) Successful face candidate area selection and face detection by the proposed face detection model, respectively

One of aims of the proposed model is to improve face detection speed by reducing the searching regions using the selective attention model before conducting face detection by the AdaBoost. As shown in Table1, the proposed model can successfully find human faces in time within 0.0539~0.2624 sec. Table 1 compares the processing time of the proposed model with that of the conventional AdaBoost model provided by OpenCV. We considered both frontal faces and profile faces with three classifiers.

The experiments were conducted for UCD database obtained in indoor environments [13]. UCD database has 530 facial images and all the image size is 360 by 288. As shown in Table 1, the proposed model shows better performance in terms of both the face detection processing speed and face detection accuracy than those of the conventional AdaBoost model.

Table 1. Face detection speed and accuracy comparison between the proposed model and the AdaBoost method based on Haar-like form features provided by OpenCV

	Proposed Model	Conventional ADABOOST
Saliency Map Processing time	35.7 ms ~ 60.8 ms	None
ADABOOST Processing time	7.75 ms ~ 240.1 ms	199.8 ms ~ 263.9 ms
Total Processing time	53.9 ms ~ 262.4 ms	206.4 ms ~ 270.8 ms
True Positive	100%	100%
False Positive	3%	8.4%

4 Conclusion

We proposed the face detection model by simply imitating human early visual mechanisms in order to localize the human face areas by combining an AdaBoost algorithm in real time. In natural complex scenes, the proposed model not only successfully localizes the face areas but also appropriately rejects non-face areas. The proposed model is based on the face color related features in order to generate face color preferable attention and the AdaBoost algorithm based on Haar-like features decides whether the attended region contains a face characteristic. The proposed

model aims to enhance the conventional AdaBoost algorithm based face detection model by considering the advantages of biologically motivated vision system.

Even though the proposed model could give plausible results to make human face selective regions, as further works we need to verify the performance of the proposed model through comparison study with that of the other models using more complex benchmark databases.

References

1. Yang, M., Kriegman, D.J., Ahuja, N.: Detecting faces in images: a survey. *IEEE Trans. Patt. Anal. Mach. Intell.* 24(1), 34–58 (2002)
2. Viola, P., Jones, M.J.: Robust real-time face detection. *International Journal of Computer Vision* 57(2), 137–154 (2004)
3. Walther, D., Itti, L., Riesenhuber, M., Poggio, T., Koch, C.: Attentional selection for object recognition – a gentle way. In: Bühlhoff, H.H., Lee, S.-W., Poggio, T.A., Wallraven, C. (eds.) *BMCV 2002. LNCS*, vol. 2525, pp. 472–479. Springer, Heidelberg (2002)
4. Serre, T., Riesenhuber, M., Louie, J., Poggio, T.: On the role of object-specific features for real world object recognition in biological vision. In: Bühlhoff, H.H., Lee, S.-W., Poggio, T.A., Wallraven, C. (eds.) *BMCV 2002. LNCS*, vol. 2525, pp. 387–397. Springer, Heidelberg (2002)
5. Navalpakkam, V., Itti, L.: An integrated model of top-down and bottom-up attention for optimal object detection. In: *CVPR*, pp. 2049–2056 (2006)
6. Siagian, C., Itti, L.: Biologically-inspired face detection: Non-Brute-Force-Search approach, 2004. In: *CVPRW 2004*, Washington, DC, USA, vol. 5, pp. 62–69 (2004)
7. Ban, S.W., Lee, M., Yang, H.S.: A face detection using biologically motivated bottom-up saliency map model and top-down perception model. *Neurocomputing* 56, 475–480 (2004)
8. Schiller, P.H.: Area V4 of the primary visual cortex. *American Psychological Society* 3(3), 89–92 (1994)
9. Goldstein, E.B.: *Sensation and perception*, 4th edn. An international Thomson publishing company, USA (1996)
10. Park, S.J., An, K.H., Lee, M.: Saliency map model with adaptive masking based on independent component analysis. *Neurocomputing* 49, 417–422 (2002)
11. Kovač, J., Peer, P., Solina, F.: Human skin colour clustering for face detection. *EUROCON 2*, 144–148 (2003)
12. Otsu, N.: A threshold selection method from gray-level histogram. *IEEE Trans. System Man Cybernetics*, 62–66 (1979)
13. UCD Valid Database, <http://ee.ucd.ie/validdb/datasets.html>

Top-Down Object Color Biased Attention Using Growing Fuzzy Topology ART

Byungku Hwang¹, Sang-Woo Ban², and Minho Lee¹

¹ School of Electrical Engineering and Computer Science, Kyungpook National University,
1370 Sankyuk-Dong, Puk-Gu, Taegu 702-701, Korea

² Dept. of Information and Communication Engineering, Dongguk University,
707 Seokjang-Dong, Gyeongju, Gyeongbuk 780-714, Korea
{bkhwang, mholee}@ee.knu.ac.kr, swban@dongguk.ac.kr

Abstract. In this paper, we propose a top-down object biased attention model which is based on human visual attention mechanism integrating feature based bottom-up attention and goal based top-down attention. The proposed model can guide attention to focus on a given target colored object over other objects or feature based salient areas by considering the object color biased attention mechanism. We proposed a growing fuzzy topology ART that plays important roles for object color biased attention, one of which is to incrementally learn and memorize features of arbitrary objects and the other one is to generate top-down bias signal by competing memorized features of a given target object with features of an arbitrary object. Experimental results show that the proposed model performs well in successfully focusing on given target objects, as well as incrementally perceiving arbitrary objects in natural scenes.

Keywords: Top-down object color biased attention, bottom-up attention, growing fuzzy topology ART.

1 Introduction

Human vision system can effortlessly detect an arbitrary object in natural or cluttered scenes, and incrementally perceive an interesting object in dynamic visual scenes. Such a visual search performance will require both bottom-up and top-down control sources to be considered and balanced against one another [1]. In Desimone and Duncan's biased competition model, the biased competition view of visual search proposed two general sources for the control of attention: bottom-up sources that arise from sensory stimuli present in a scene and top-down sources that arise from the current behavioral goals [1,2]. Itti and Navalpakkam proposed a top-down attention model that has a biasing mechanism to salient map based on signal to noise ratio of color and orientation feature generated by a bottom-up process [3]. The performance of this model seems to be decreased according to different backgrounds. Walther and Koch also proposed a top-down attention model having bias based on features generated from a bottom-up process [4]. Torralba and Oliva proposed a top-down attention model using spatial information [5], which may show poor performance for objects

located at unusual place. In this paper, therefore, we proposed a new top-down attention model that can have more robust performance in localizing a given object in a various situations with unusual placement and varying backgrounds.

Based on the biased competition mechanism, we proposed a biologically motivated top-down object biased attention model. The proposed attention model consists of two parts. One is the bottom-up attention part that can pop-out salient areas by calculating the relativity of primitive visual features such as intensity, edge, and color [6]. The other is the top-down object biased attention part. In order to generate object biased signal, we need object perception and memorization mechanism such as working memory in a human brain. In this model, we propose a new growing fuzzy topology adaptive resonance theory (TART) model for object perception and memorization that makes object color feature clusters in an incremental mode. The proposed growing fuzzy TART not only increases stability in conventional fuzzy ART while maintaining plasticity, but it also preserves topology structures in input feature spaces that are divided by color and form domains in an object. Finally, the growing fuzzy TART makes clusters in order to construct an ontology map in the color domains. The growing fuzzy TART can activate the target object related features among memorized features and compare the activated features with features of an arbitrary object in order to generate a proper top-down bias signal that can make a target colored object area in an input scene become the most salient area. Moreover, the clustered information in the growing fuzzy TART is relevant for describing specific colored objects, and thus it can automatically generate an inference for unknown objects by using learned information. Experimental results show that the proposed model properly guides color feature based top-down object biased attention.

This paper is organized as follows; Section 2 describes the proposed model combining the bottom-up saliency map (SM) model with the growing fuzzy topology ART model. The experimental results will be followed in Section 3. Section 4 presents further works, conclusions and discussions.

2 The Proposed Model

When humans pay attention to a target object, the prefrontal cortex gives a competitive bias signal, related with the target object, to the infero-temporal (IT) and the V4 area. Then, the IT and the V4 area generates target object dependant information, and this is transmitted to the low-level processing part in order to make a competition between the target object dependant information and features in every area in order to filter the areas that satisfy the target object dependant features.

Figure 1 shows the overview of the proposed model during training mode. As shown in the lower part of Fig. 1, the bottom-up attention part generates a bottom-up SM based on primitive input features such as intensity, edge and color opponency. In training mode of the proposed model, each salient object decided by bottom-up attention is learned by the growing fuzzy TART. For each object area, the log-polar transformed features of RG and BY color opponency are used as color features for representing an object. Those are used as input of the growing fuzzy TART.

In the top-down object biased attention model, which is shown in Fig. 1, the growing fuzzy TART activates the memorized color features of the target object when a

task of target object searching is being given. The activated color features related with the target object are involved in competition with the color features extracted from each bottom-up salient object area in an input scene. By such a competition mechanism, as shown in Fig. 1, the proposed model can generate a top-down signal that can bias the target object area in the input scene. Finally the top-down object biased attention model can generate a top-down object biased saliency map, in which the target object area is popped out. Therefore, the growing fuzzy TART works the most important role for the top-down object biased attention. In this model, the proposed growing fuzzy TART is implemented by integrating the conventional fuzzy ART, with the topology-preserving mechanism of the growing cell structure (GCS) unit [7]. In the growing fuzzy TART, each node in the F2 layer of the conventional fuzzy ART network was replaced with GCS units.

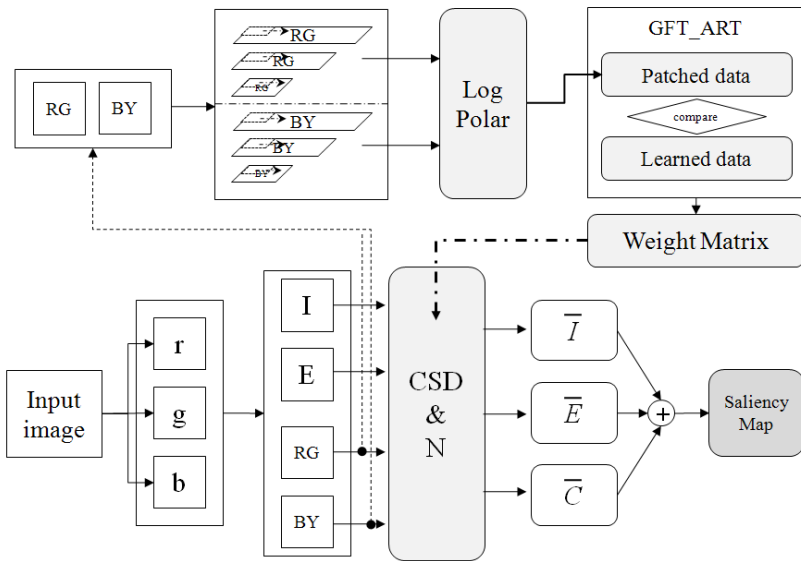


Fig. 1. Overview of the proposed model during training mode (r: red, g: green, b: blue, I: intensity feature, E: edge feature, RG: red-green opponent coding feature, BY: blue-yellow opponent coding feature, CSD&N: center-surround difference and normalization, \bar{I} : intensity feature map, \bar{E} : edge feature map, \bar{C} : color feature map, GFT_ART: growing fuzzy TART)

2.1 Bottom-Up Selective Attention Model

In the bottom-up processing, the intensity, edge and color features are extracted in the retina. These features are transmitted to the visual cortex through the lateral geniculate nucleus (LGN). While transmitting those features to the visual cortex, intensity, edge, and color feature maps are constructed using the on-set and off-surround mechanism of the LGN and the visual cortex. And those feature maps make a bottom-up SM model in the lateral intral-parietal cortex (LIP) [8].

In order to implement a human-like visual attention function, we consider the simplified bottom-up SM model [9]. In our approach, we use the SM model that reflects the functions of the retina cells, the LGN and the visual cortex. Since the retina cells can extract edge and intensity information as well as color opponency, we use these factors as the basic features of the SM model [8, 9]. And the real color components R, G, B, Y are extracted using normalized color coding [8].

Actually, considering the function of the LGN and the ganglian cells, we implement the on-center and off-surround operation by the Gaussian pyramid images with different scales from 0 to n -th level, whereby each level is made by the sub-sampling of 2^n , thus it is able to construct four feature basis such as the intensity (I), and the edge (E), and color (RG and BY) [9, 10]. This reflects the non-uniform distribution of the retina-topic structure. Then, the center-surround mechanism is implemented in the model as the difference operation between the fine and coarse scales of the Gaussian pyramid images [9, 10]. Consequently, the three feature maps such as \bar{I} , \bar{E} and \bar{C} can be obtained by the center-surround difference algorithm [9]. A SM is generated by the summation of these three feature maps. The salient areas are obtained by searching a maximum local energy with a fixed window size shifting pixel by pixel in the SM.

2.2 Growing Fuzzy TART

Figure 2 shows the proposed growing fuzzy TART.

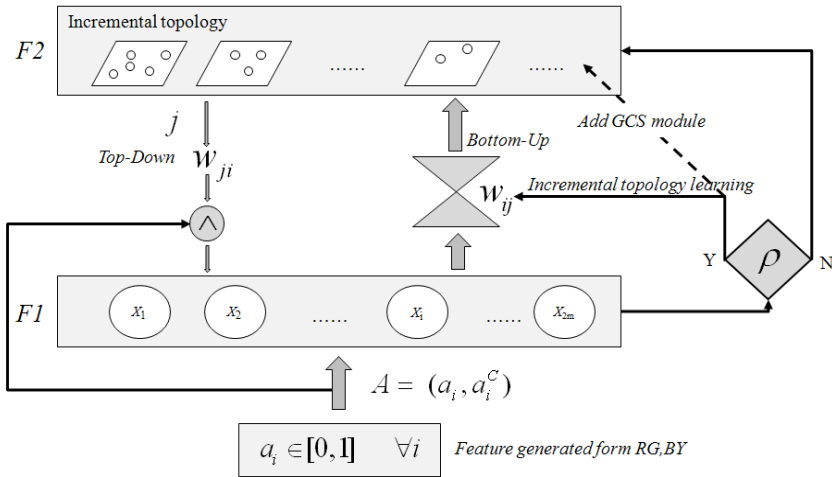


Fig. 2. Growing fuzzy topology adaptive resonance theory (TART) network

The inputs of the growing fuzzy TART consist of the color features. These features are normalized and then represented as a one-dimensional array X that is composed of every pixel value a_i of the color features and each complement a_i^c is calculated by

$1 - a_i$, the values of which are used as an input pattern in the F1 layer of the growing fuzzy TART model. Next, the growing fuzzy TART finds the winning GCS unit from all GCS units in the F2 layer, by calculating the Euclidean distance between the bottom-up weight vector W_i , connected with every GCS unit in the F2 layer, and X is inputted. After selecting the winner GCS unit, the growing fuzzy TART checks the similarity of input pattern X and all weight vectors W_i of the winner GCS unit. This similarity is compared with the vigilance parameter ρ , which is the minimum these results similarity between the input pattern and the winner GCS.

If the similarity is larger than the vigilance value, a new GCS unit is added to the F2 layer. In such situation, resonance has occurred, but if the similarity is less than the vigilance, the GCS algorithm is applied. The detailed GCS algorithm is described as the following [7]:

For initialization, one GCS unit in the F2 layer is created with three nodes n_1, n_2, n_3 for its topology and randomly initialized the weight W_i . C , as the connection set, is defined as the empty set $C = \emptyset$. A is the set of nodes in a GCS unit.

For each node i in the network, the GCS calculates the distance from the input $\|X - W_i\|$. The GCS selects the best-matching node and the second best, that are nodes s and $t \in A$, such that

$$s = \arg \min_{n \in A} \| \xi - W_n \| \quad (1)$$

$$t = \arg \min_{n \in A / \{s\}} \| \xi - W_n \| \quad (2)$$

where W_n is the weight vector of node n . If there is no connection between s and t , a connection is created between s and t .

$$a = \exp(-\|X - W_s\|) \quad (3)$$

If activity a is less than activity threshold a_T , a new node should be added between the two best-matching nodes, s and t . First, GCS adds the new node r

$$A = A \cup \{(r)\} \quad (4)$$

GCS creates the new weight vector by, setting the weights to be the average of the weights for the best matching node and the second best node

$$W_r = (W_s + W_t) / 2 \quad (5)$$

Edges are inserted between r and s and between r and t

$$C = C \cup \{(r, s), (r, t)\} \quad (6)$$

The link between s and t is removed, which is denoted as Eq. (7).

$$C = C / \{(s, t)\} \quad (7)$$

The weight of the winning node, W_s is adapted by Eq. (8) and the weight of its neighborhood node, W_i , is adapted by Eq. (9), where \mathcal{E}_b and \mathcal{E}_n are the training parameters for the winner node and the neighborhood node, respectively.

$$W_s = \mathcal{E}_b * (\xi - W_s) \quad (8)$$

$$W_i = \mathcal{E}_n * (\xi - W_i) \quad (9)$$

Our approach hopefully enhances the dilemma regarding the stability of fuzzy ART and the plasticity of GCS [7, 11]. The advantages of this integrated mechanism are that the stability in the convention fuzzy ART is enhanced by adding the topology preserving mechanism in incrementally-changing dynamics by the GCS, while plasticity is maintained by the fuzzy ART architecture. Also, adding GCS to fuzzy ART is good not only for preserving the topology of the representation of an input distribution, but it also self adaptively creates increments according to the characteristics of the input features.

3 Experimental Results

Figure 3 shows the simulation results of the proposed top-down object biased attention model after trained a yellow ball. As shown in a bottom-up SM result image in Fig. 3, the yellow colored object, is not mostly salient area but the 2nd salient area when based on only bottom-up features without considering top-down bias. However, the yellow colored object became the most salient area after considering top-down bias in conjunction with bottom-up attention as shown in the top-down object biased final saliency areas result image in Fig. 3.

Table 1 shows the performance of the proposed model. We conducted experiments for two different object image DB [12, 13]. One is from ABR Lab. in KNU and the other is from Itti's Lab. The proposed model learned the red and yellow colors from randomly chosen 10 red and yellow objects. Table 1 shows the performance for testing red and yellow color biasing of the proposed model after training the red and yellow colors. As shown in Table 1, the proposed model shows successful color biasing performance by 82.5% for the object image DB of ABR Lab for which illumination status is known. However, the proposed model shows 70% for the object image DB of Itti's Lab for which there is no information about illumination status. Therefore we need to develop more plausible biasing model for showing better performance for general object image DBs. In the proposed model, the vigilance parameter value affects the number of nodes generating in F2 layer, which also affect performance of the model. Table 2 shows experimental results comparing the correct top-down biasing performance according to the different vigilance parameter values.

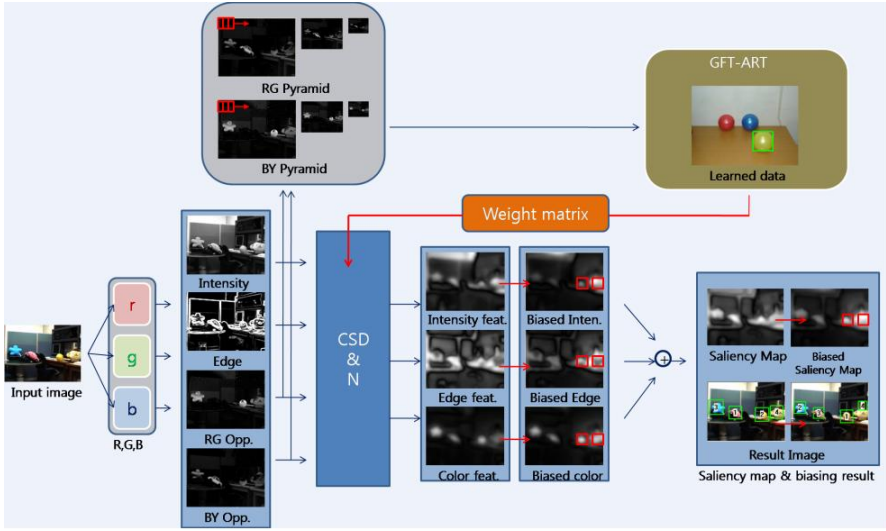


Fig. 3. Top-down biased attention results of the proposed model

Table 1. Experimental results of the proposed model for two object image databases of Itti’s Lab. image DB and ABR Lab. image DB (after training red and yellow colors obtained from 10 randomly chosen images)

Image Database	Correctly biasing images	Incorrectly biasing images	Correctness
Itti’s Lab. DB (40 object images)	28	12	70%
ABR Lab. DB (40 object images)	33	7	82.5%

Table 2. Comparison of top-down biasing performances of the proposed model according to varying vigilance parameter values. (ABR Lab. Image Database: 40 images).

Vigilance parameter value	# of correctly biasing images	# of incorrectly biasing images	Correctness
0.8	17	23	42.5%
0.9	33	7	82.5%
0.95	15	25	37.5%

4 Conclusion

In conclusion, we proposed a biologically motivated selective attention model that can provide a proper visual object search performance based on top-down biased

competition mechanism considering both spatial attention and goal oriented object color biased attention. In the proposed model, we only considered the color features for top-down biased attention. As a further work, we are now considering the form features for top-down biased attention.

Acknowledgments. This research was funded by the Brain Neuroinformatics Research Program of the Ministry of Commerce, Industry and Energy in Korea.

References

1. Vecera, S.P.: Toward a biased competition account of object-based segregation and attention. *Brain and Mind* 1, 353–384 (2000)
2. Desimone, R., Duncan, J.: Neural mechanisms of selective visual attention. *Annual Review of Neuroscience* 18, 193–222 (1995)
3. Navalpakkam, V., Itti, L.: An integrated model of top-down and bottom-up attention for optimal object detection. In: *CVPR 2006*, pp. 2049–2056 (2006)
4. Walther, D., Koch, C.: Modeling attention to salient proto-objects. *Neural Networks* 19(9), 1395–1407 (2006)
5. Torralba, A., Oliva, A., Castelhano, M., Henderson, J.M.: Contextual guidance of attention in natural scenes: The role of global features on object search. *Psychological Review* 113(4), 766–786 (2006)
6. Won, W.J., Yeo, J., Ban, S.W., Lee, M.: Biologically motivated incremental object perception based on selective attention. *Int. J. Pattern Recognition & Artificial Intelligence* 21(8), 1293–1305 (2007)
7. Marsland, S., Shapiro, J., Nehmzow, U.: A self-organising network that grows when required. *Neural Networks, Special Issue* 15(8-9), 1041–1058 (2002)
8. Goldstein, E.B.: *Sensation and perception*, 4th edn. An international Thomson publishing company, USA (1996)
9. Park, S.J., An, K.H., Lee, M.: Saliency map model with adaptive masking based on independent component analysis. *Neurocomputing* 49, 417–422 (2002)
10. Choi, S.B., Jung, B.S., Ban, S.W., Niitsuma, H., Lee, M.: Biologically motivated vergence control system using human-like selective attention model. *Neurocomputing* 69, 537–558 (2006)
11. Carpenter, G.A., Grossberg, S., Makuzon, N., Reynolds, J.H., Rosen, D.B.: Fuzzy ART-MAP: A neural network architecture for incremental supervised learning of analog multi-dimensional maps. *IEEE Transactions on Neural Networks* 3(5), 698–713 (1992)
12. ABR Lab. Image database, <ftp://abr.knu.ac.kr>
13. Itti's Lab. Image database, <http://ilab.usc.edu/research>

A Study on Human Gaze Estimation Using Screen Reflection

Nadeem Iqbal and Soo-Young Lee

Computational NeuroSystems Lab
Department of Bio & Brain Engineering
KAIST, Daejeon. 305-701, Republic of Korea
nadeem@neuron.kaist.ac.kr, sylee@kaist.ac.kr

Abstract. Many eye gaze systems use special infrared (IR) illuminator and choose IR-sensitive CCD camera to estimate eye gaze. The IR based system has the limitation of inaccurate gaze detection in ambient natural light and the number of IR illuminator and their particular location has also effect on gaze detection. In this paper, we present a eye gaze detection method based on computer screen illumination as light emitting source and choose high speed camera for image acquisition. In order to capture the periodic flicker patterns of monitor screen the camera is operated on frame rate greater than twice of the screen refresh rate. The screen illumination produced a mark on the corneal surface of the subject's eye as screen-glint. The screen reflection information has two fold advantages. First, we can utilize the screen reflection as screen-glint, which is very useful to determine where eye is gazing. Secondly the screen-glint information utilize to localized eye in face image. The direction of the user's eye gaze can be determined through polynomial calibration function from the relative position of the center of iris and screen-glint in both eyes. The results showed that our propose configuration could be used for gaze detection method and this will lead to increased gaze detection role for the next generation of human computer interfaces.

Keywords: Human Computer Interaction, Gaze Estimation, Screen-glint, and Screen Reflection.

1 Introduction

Image processing and computer vision techniques play an important role for designing and understanding research in vision based technologies. The use of robots, computer and vision based interface in our life provide a valuable stimulus for designing a better HCI (Human Computer Interaction). Recently many friendly technologies facilitated with multimodal HCI approach. The Eye gaze detection plays a pivotal role in the HCI model.

Current eye gaze interaction models use special infrared (IR) illumination as source of illumination varying from one IR illuminator to many according to the application. For the acquisition of the IR images the system uses IR sensitive CCD camera having different focal length depends on application. Such IR-based models are used in too many applications.

Hutchinson's [1] used corneal reflection (glint) and pupil image (bright eye's) from the eyes image obtained from video camera located immediately below the center of computer, while irradiating the eye with invisible infrared light using light emitting diode(LED) positioned in front of the eye. Ebisawa's [2] used two light sources and image difference method. One light source which is coaxial with camera optical axis produce bright pupil image against the darker background called the bright condition. Zhu's [3] gaze model uses two ring structures. The IR LED is mounted in a circular shape. The eye is illuminated with both rings. The outer ring is located off axis and the inner ring is on the optical axis to generate dark and bright pupil effect. Yoo's [4] model require five light sources and two CCD cameras . Four light sources are attached to the corners of a monitor to cause the reflections on the surface of the cornea and the other one is located at the center of camera lens to make bright eye effect. However, IR based system has the limitation of inaccurate gaze detection in ambient natural light. Due to this reason its application is widely restricted in many application areas which operate in natural light. In most gaze estimation model the location and number of IR sources are also influence on the result of gaze estimation.

An alternative configuration for eye gaze estimation proposed with single high speed camera. In order to illuminate the eye we used screen illumination as light emitting source. Which provide a screen glint (cornea reflection) , The screen-glint work as glint in the previous research to estimate the eye gaze . The method of iris detection is utilized to find the center of iris. The direction of eye gaze calculated from the relative position of the center of iris and screen glint in both eyes.

In section 2, we present the proposed configuration and discuss the eye tracking algorithm. Section 3 is about iris segmentation. Experimental results and conclusion are included in section 4 and 5 respectively.

2 Proposed Configuration

In proposed configuration the monitor is placed on the desk and user is sit about 60 cm in front of the monitor and the camera is mounted at the center beneath of the screen. The screen illumination (screen flicker pattern) is use to generate corneal reflection. In order to capture the screen flicker pattern we have used high speed camera (IPX-VGA-210-L) which capture the image with frame rate greater than twice of the screen refresh rate. In our experiment the screen refresh rate is 60 Hz and camera speed operate with speed of 120 frames per second. The resolution of capture image is 640 x 480. The screen illumination is reflected off the corneal surface and appears in the camera as a small intense area which we call as "screen-glint".



Fig. 1. Shape of screen glint in eye region

The shape of the screen glint seems like rectangular shape which is akin with monitor screen as shown in Fig.1. The screen illumination generates a flicker pattern which comes from when electron beam is moved from bottom of screen display to the top of screen display. Two consecutive images have produced the full movement of the electron beam of the screen display. The reflection of these patterns at each frame at eye region shows in Fig. 2.

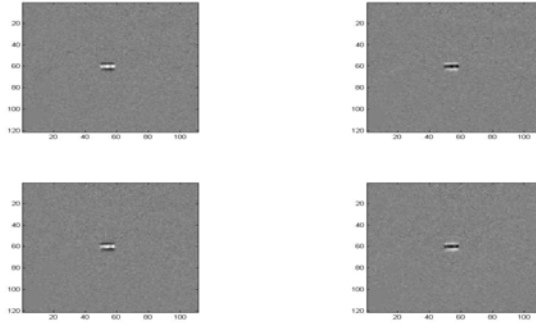


Fig. 2. Screen reflection pattern in the eye region

2.1 Detection of Screen-Glint Center

First, we calculate the difference between 5 consecutive frames and find the aggregative sum. The resultant image is shown in Fig.4. We continue this procedure for all images with center and surround approach. Second, find the localized eye region based on the screen glint location at each eye in the image, which is described in section 2.2. Third, compute the rough center of each screen-glint polygon/rectangle by finding maximum intensity location in resultant image. Fourth, find the boundary point of screen glint by 1-D line edge search along the normal at center pixel. Finally, after knowing the height and width of the screen-glint refine/recalculate the center of screen-glint.

2.2 Eye Localization Method

Our proposed system utilizes the high camera frame rate to capture the screen refresh rate in the eye image. We observe that the full movement of screen flicker pattern obtain in two consecutive images. We calculate the difference image by subtracting these two consecutive images. In order to capture good reflection pattern we use five difference images summation. To remove noisy pixels we use low-pass filter.

In the proposed method we first find maximal pixel intensity (high peak) at resultant difference image. The resultant region of interests ROI around first high peak is considered first candidate eye region. After marking the location of first eye region, we keep first eye region to zero for finding second eye region this will also suppress all high peak of first region.

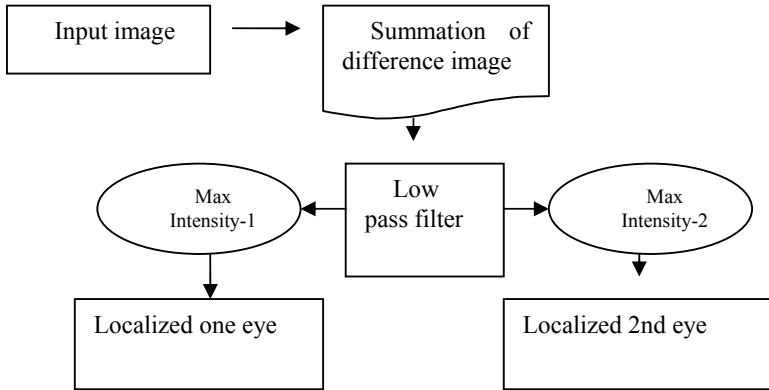


Fig. 3. Algorithm for eye localization

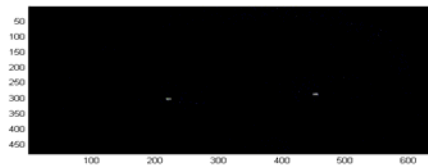


Fig. 4. Resultant Difference image

Similarly, we have calculated the second high intensity region (high peak). Therefore we find two maximum points to localize two eye regions as show in Fig.5.

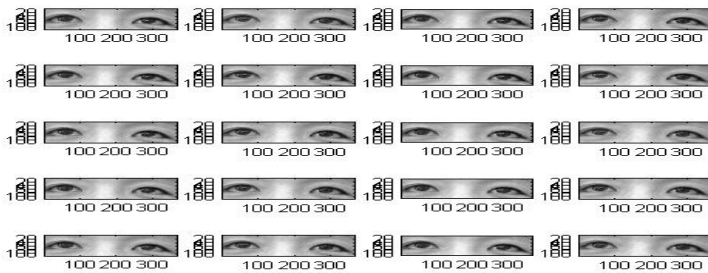


Fig. 5. Localized eye region

3 Center of Iris Detection

Iris segmentation is one of the most important task for the eye gaze estimation. The eye gaze estimation accuracy is proportionally dependable on the accurate iris center. Inaccurate iris center will make large estimation error. For iris segmentation we utilized the contrast between sclera and iris region. The iris segmentation algorithm uses

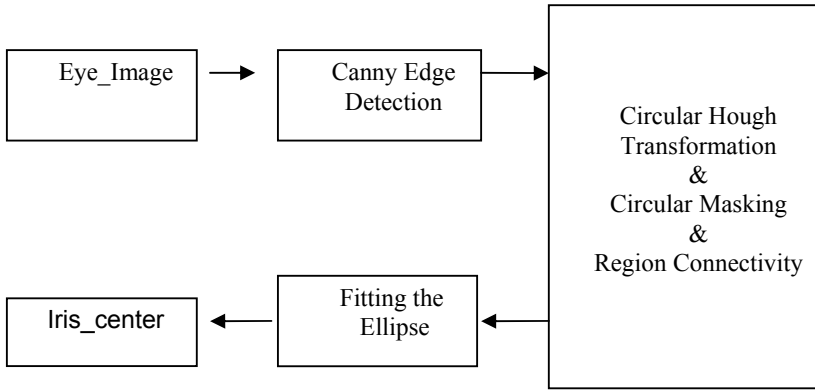


Fig. 6. Iris segmentation algorithm

each eye region separately and finally combines the center of both iris. The edge map image of the original eye image is computed using canny edge detection followed by a circular Hough transformation [5] in order to robustly detect the circle in the edge map image.

In order to find refine arc of the iris for fitting ellipse we use two circular masking and region connectivity approach. The circular masking uses two circular filter .First circular filter have larger radius of the detected circle to remove outer region edges which include the corner of the eyes. Second circular filter have smaller radius than detected circle and use to remove the inner circular region edge which include eyelids. We have also applied region connectivity filter which removed non-arc iris boundary and noise. Once we find the proper arc of the iris boundary we used the direct fitting ellipse algorithm [6] to estimate the iris center. The result of iris segmentation is shown in Fig. 7.

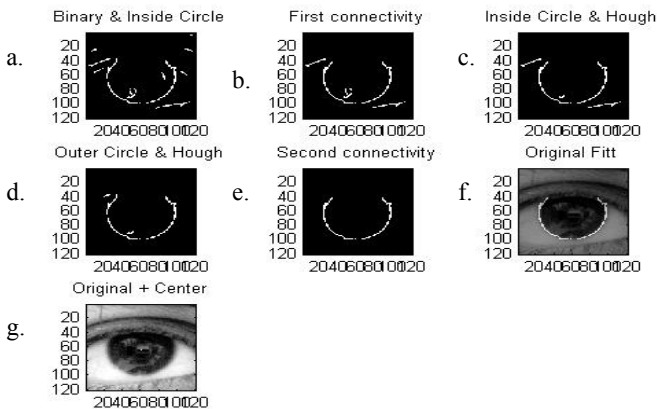


Fig. 7. a) Edge Image b) Region Connectivity c) Hough Inner Circle d) Hough Outer Circle e) Second Region Connectivity f) Iris edge on the original image. g) Center of iris.

4 Experiments and Results

To analyze the gaze estimation with proposed configuration we have used second order polynomial equation as mapping function between screen coordinate and gaze vector through calibration [7]. The center of iris and the center of screen glint define a vector in the image. We have obtained two such vectors from left and right eye. These vectors can be easily mapped to the screen coordinate on a computer monitor after a calibration. The mapping from these vectors to screen coordinate is performed using a pair of second order polynomial as

$$s_x = a_0 + a_1 x_{ig} + a_2 y_{ig} + a_3 x_{ig} y_{ig} + a_4 x_{ig}^2 + a_5 y_{ig}^2 \quad (1)$$

$$s_y = b_0 + b_1 x_{ig} + b_2 y_{ig} + b_3 x_{ig} y_{ig} + b_4 x_{ig}^2 + b_5 y_{ig}^2 \quad (2)$$

Where (s_x, s_y) are the screen coordinates and (x_{ig}, y_{ig}) is the iris-screen glint vector. In our simulation we used 9 points for calibration to estimate the parameters a_0 - a_5 and b_0 - b_5 using least square [8]. During the calibration, there is no head movement and the user is gaze to each point. After knowing the calibration parameters new data has been tested for the said users with no head movement. The distance between user and CRT screen monitor is 60 cm. The resolution of image is 640 x 480 having high speed camera with zoom lens of 25mm. In experiment, we capture both eyes of the user. The system utilizes each eye feature and gaze estimation individually then finally combined the average results for each eye gaze estimation.

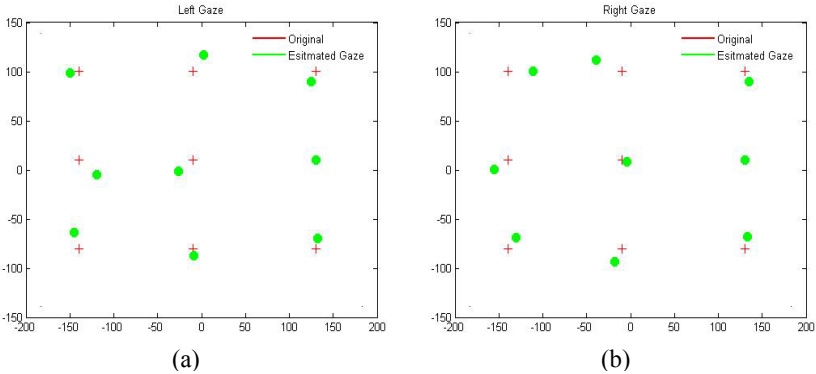


Fig. 8. Gaze detection result when using proposed configuration. a) Left eye gaze estimation b) Right eye gaze Estimation.

In order to combine the results of both left and right eye we have analyzed two approaches. In first approach, we have calculated the average of left and right gaze vector and then estimated the gaze position through mapping function. In second approach, we have calculated the gaze vector for each eye then estimated gaze

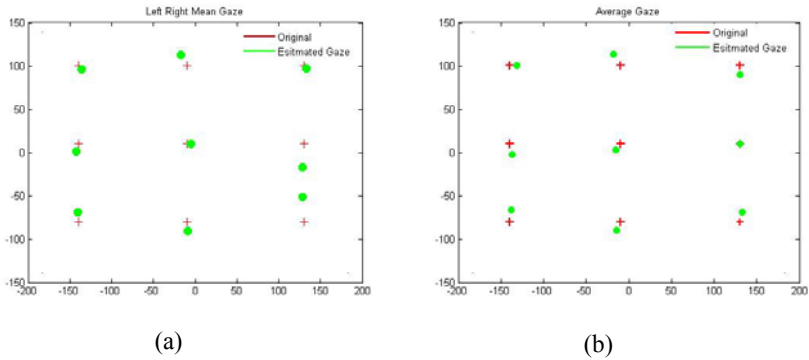


Fig. 9. Both Eye Result. a) Average of Gaze Vector (Approach-1) b) Average of Gaze position (Approach-2).

position through mapping function for each eye and the average the gaze estimation result. As shown in Fig.9.

The result indicates that using both eye information can achieve better gaze estimation as compare to the gaze estimation of single eye. The human also utilizes both eye to gaze her/his eyes on the subject.

5 Conclusion and Future Works

In this paper an eye gaze detection method with screen reflection has studied, our system used one high speed camera and to generate the cornea reflection we utilized the screen illumination as source of illumination to illuminate the eye. The system is using information of relative position of iris center and screen glint of both eyes to determine the gaze detection. The results suggest that both eye have achieved better performance then single eye. The result shows that propose configuration can be used as gaze detection method which will provide more flexible and lead to increase the gaze detection role in HCI applications. In future, we plan to measure gaze detection in various environments with sufficient head movement.

Acknowledgments. This research was supported by a scholarship grant of the Korea government IT scholarship program (IITA).

References

1. Hutchinson, T.E., White Jr., K.P., Reichert, K.C., Frey, L.A.: Human-computer interaction using eye gaze point. *IEEE Trans.Syst., Man, Cybern.* 19, 1527–1533 (1989)
2. Ebisawa, Y.: Improved video based eye-gaze detection method. *IEEE Trans. On Instrument and Measurement* 47(4), 476–480 (1998)
3. Zhu, Z., Ji, Q.: Eye and gaze tracking for interactive graphic display. *Machine Vision and Application* 15, 139–148 (2004)

4. Yoo, D.H., Chung, M.J.: A novel non-intrusive eye gaze estimation using cross-ratio under large head motion. *Computer Vision and Understanding* 98, 25–52 (2005)
5. Wildes, R., Asmuth, J., Green, G., Hsu, S., Kolczynski, R., Mately, J., McBride, S.: A system for automated iris recognition. In: *Proceedings IEEE Workshop on Applications of Computer Vision*, Sarasota, FL, pp. 121–128 (1994)
6. Fitzgibbon, A., Pilu, M., Fisher, R.B.: Direct Least Square Fitting of Ellipses. *IEEE transaction Pattern Analysis and Machine Intelligence* 21(5) (May 1999)
7. Morimot, C.H., Mimica, R.M.: Eye gaze tracking techniques for interactive applications. *Computer Vision and Image Understanding* 8, 4–24 (2005)
8. Ramdane-Cherif, Z., Nait, A., Motsch, J.F., Krebs, M.O.: Performance of computer system for recording and analyzing eye gaze position using an infrared light device. *Journal of clinical Monitoring and Computing* 18, 39–44 (2004)
9. Ohno, T., Mukawa, N., Yoshikawa, A.: FreeGaze.: A gaze tracking system for everyday gaze interaction. In: *Proceeding of Eye Tracking Research & Application Symposium*, pp. 125–132 (2002)

A Novel GA-Taguchi-Based Feature Selection Method

Cheng-Hong Yang¹, Chi-Chun Huang², Kuo-Chuan Wu¹, and Hsin-Yun Chang³

¹ Department of Computer Science and Information Engineering,
National Kaohsiung University of Applied Sciences
415 Chien Kung Road, Kaohsiung 80778, Taiwan
chyang@cc.kuas.edu.tw, kuo.chuan.wu@gmail.com

² Department of Information Management,
National Kaohsiung Marine University
142 Hai Jhuan RD., Nanzih District, Kaohsiung 811, Taiwan
cchuang@mail.nkmu.edu.tw

³ Department of Business Administration, Chin-Min Institute of Technology
110 Hsueh-Fu Road, Tou-Fen, Miao-Li 305, Taiwan
Department of Industrial Technology Education, National Kaohsiung Normal University
116 Heping 1st RD., Lingya District, Kaohsiung 802, Taiwan
ran_hsin@ms.chinmin.edu.tw

Abstract. This work presents a novel GA-Taguchi-based feature selection method. Genetic algorithms are utilized with randomness for “global search” of the entire search space of the intractable search problem. Various genetic operations, including crossover, mutation, selection and replacement are performed to assist the search procedure in escaping from sub-optimal solutions. In each iteration in the proposed nature-inspired method, the Taguchi methods are employed for “local search” of the entire search space and thus can help explore better feature subsets for next iteration. The two-level orthogonal array is utilized for a well-organized and balanced comparison of two levels for features—a feature is or is not selected for pattern classification—and interactions among features. The signal-to-noise ratio (SNR) is then used to determine the robustness of the features. As a result, feature subset evaluation efforts can be significantly reduced and a superior feature subset with high classification performance can be obtained. Experiments are performed on different application domains to demonstrate the performance of the proposed nature-inspired method. The proposed hybrid GA-Taguchi-based approach, with wrapper nature, yields superior performance and improves classification accuracy in pattern classification.

Keywords: Genetic Algorithm, Taguchi Method, Orthogonal Array, Feature Subset Selection, Pattern Classification.

1 Introduction

Over the past decade, different pattern classification approaches have been applied to classify new, unseen instances. In a pattern classification framework [8], a set of training instances or examples, denoted as training set *TS*, is given. Each instance or

example consists of n features and a class label. All features of each instance are generally considered during the classification process. Various real-world pattern classification problems, however, involve unimportant or irrelevant features that typically have a marked effect on overall classification accuracy. To improve classification performance, many feature selection and feature subset selection methods have been investigated [2][6][7][10][12][13][14][15][16][17]. These approaches focus on selecting important and relevant features from an original feature set and reducing the dimensionality in a particular pattern classification problem.

Feature subset selection can be considered a search problem [12]. Each search state in the search space defines a possible feature subset. If each instance in a specific classification problem has n attributes, the search space is formed by 2^n possible or candidate feature subsets. Clearly, an exhaustive search of the entire search space (i.e., 2^n possible feature subsets) has a very high computational cost and, thus, is generally unfeasible in practice, even for a medium-sized p [17]. Consequently, selecting a best feature subset for pattern classification from the entire search space is very difficult, in particular with regard to the tradeoff between high classification accuracy and small number of selected features.

This paper presents novel hybrid GA-Taguchi-based feature selection method. Genetic Algorithms [9][18] are utilized with randomness for “global search” of the entire search space (i.e., 2^n possible feature subsets). Various genetic operations, including crossover, mutation, selection and replacement are performed to assist the search procedure in escaping from sub-optimal solutions [17]. In each iteration in the proposed nature-inspired method, the Taguchi methods [20][21] are utilized to help explore better feature subsets (or solutions), which are somewhat different from those in candidate feature subsets, for the next iteration. In other words, the Taguchi methods are employed for “local search” of the entire search space (i.e., 2^n possible feature subsets). Consider that two feature subsets or solutions, b_1 and b_2 , which are selected from a set of candidate feature subsets, have w different bits ($w \leq n$). The two-level orthogonal array, a central concept of the Taguchi method, is utilized in the proposed scheme for a well-organized and balanced comparison of two levels for the w features—a feature is or is not selected for pattern classification—and interactions among all features in a specific classification problem. The signal-to-noise ratio (SNR) is then employed to determine the robustness of the w features. Consequently, a superior candidate feature subset, which has high classification performance for the classification task, can be obtained by considering each feature with a specific level having a high signal-to-noise ratio (SNR). If a particular target (e.g. process or product) has d different design factors, 2^d possible experimental trials will need to be considered in full factorial experimental design. Orthogonal arrays are principally utilized to decrease experimental efforts associated with these d design parameters. Finally, prior to the classification process, feature subset evaluation efforts can be significantly reduced based on the two-dimensional, fractional factorial experimental design matrix. In this manner, important and relevant features can be identified from an original feature set for pattern classification.

The remainder of this paper is organized as follows. Section 2 reviews the concepts in Taguchi methods and the nearest neighbor rule employed in the proposed method. Section 3 presents the novel hybrid GA-Taguchi-based method for feature subset selection. In Section 4, an example is utilized to illustrate the proposed method. In

Section 5, experiments using on different classification problems are discussed. Finally, Section 6 presents conclusions.

2 Genetic Algorithms and Taguchi Methods

2.1 Genetic Algorithms

A genetic algorithm (GA) [9], also known as a stochastic search method, is based on natural selection in biological evolution [11]. Consider that a specific problem domain has a solution space with a set of possible solutions (also named individuals or chromosomes in biological evolution). A genetic algorithm attempts to find the optimal or sub-optimal solutions in the solution space based on various genetic operations, including solution (or chromosome) encoding, crossover, mutation, fitness evaluation, selection and replacement.

For a feature subset selection problem, a possible solution in the solution space is a specific feature subset that can be encoded as a string of n binary digits (or bits). Here, each feature is represented by a binary digit with values 1 and 0, which identify whether the feature is selected or not selected in the corresponding feature subset, respectively. This process is called solution (or chromosome) encoding. For instance, a string of ten binary digits (i.e., a solution or a chromosome), say, 0100100010, means that features 2, 5, and 9 are selected in the corresponding feature subset. In the first step of a general GA, some solutions are randomly selected from the solution space as the initial set CS of candidate solutions. The number of solutions in CS is denoted as the population size. When two parent solutions, p_1 and p_2 , are selected from CS , the crossover operation will be applied to generate corresponding offspring q . In other words, each feature i of offspring q is the same as either that of p_1 or that of p_2 . Consequently, the mutation operation is utilized to perturb offspring q slightly. Once a perturbed offspring is obtained, an evaluation criterion is applied to analyze the fitness of parent solutions and corresponding offspring. For each iteration (or generation), solutions that have high fitness are retained in the candidate set CS (i.e., CS is updated). These genetic operations, including crossover, mutation, fitness evaluation, selection and replacement, are repeated until a predefined number of iterations (or generations) or a particular stop condition is met. Finally, a set of optimal or sub-optimal solutions for a specific problem domain are obtained. As a result, GAs have been widely applied in many areas, such as for solving optimization problems [9]. In the proposed method, GAs are utilized with randomness for “global search” of the entire search space. Restated, the genetic operations are performed to assist the search procedure in escaping from sub-optimal solutions [17].

2.2 Taguchi Methods

In robust experimental design [20][21], processes or products can be analyzed and improved by altering relative design factors. As a commonly-used robust design approach, the Taguchi method [20][21] provides two mechanisms, an orthogonal array and signal-to-noise ratio (SNR), for analysis and improvement. If a particular target (e.g. process or product) has d different design factors, 2^d possible experimental trials will need to be considered in full factorial experimental design. Orthogonal

arrays are principally utilized to decrease experimental efforts associated with these d design parameters. An orthogonal array can be considered a fractional factorial experimental design matrix that provides a comprehensive analysis of interactions among all design factors and fair, balanced and systematic comparisons of different levels (or options) of each design factor. In this two-dimensional array, each column indicates a specific design parameter and each row represents an experimental trial with a particular combination of different levels (or options) for all design factors. The proposed scheme uses the commonly-used two-level orthogonal array for selecting representative features from the original feature set. A general two-level orthogonal array can be defined as

$$L_h(2^d), \quad (1)$$

where d is the number of columns (i.e., number of design parameters) in the orthogonal matrix, $h = 2^k$ ($h > d$, $k > \log_2(d)$ and k is an integer) denotes the number of experimental trials, base 2 denotes the number of levels (or options) of each design parameter (i.e., levels 0 and 1).

For instance, for a particular target that has 15 design parameters with two levels (i.e., levels 0 and 1), a two-level orthogonal array $L_{16}(2^{15})$ can be generated (as shown in Table 1). In this two-level orthogonal array, only 16 experimental trials are required for evaluation, analysis and improvement. Conversely, all possible combinations of 15 design factors (i.e., $2^{15}=32768$) should be considered in the full factorial experimental design, which is frequently inapplicable in practice.

Once an orthogonal array is generated, an observation or objective function of each experimental trial can be determined. The signal-to-noise ratio (SNR) is then utilized to analyze and optimize design parameters for the particular target. Generally, two signal-to-noise ratio (SNR) types, the smaller-the-better and larger-the-better types [21], are typically utilized. The signal-to-noise ratio (SNR) is utilized to determine the robustness of all levels of each design parameter. That is, “high quality” of a particular target can be achieved by specifying each design parameter with a specific level having a high signal-to-noise ratio (SNR).

3 A Novel GA-Taguchi-Based Feature Selection Method

In this section, a novel GA-Taguchi-based feature selection method is presented. Consider that a specific classification problem involves a set of m labeled training examples, defined by $T = \{t_1, t_2, \dots, t_m\}$. Each example has n features, defined by $F = \{f_1, f_2, \dots, f_n\}$. As mentioned, the search space S is composed of 2^n candidate feature subsets. In the proposed method for feature subset selection, each specific feature subset (i.e., a possible solution in S) is encoded as a string of n binary digits (or bits). Each feature is represented by a binary digit (bit) with values 1 and 0, which identify whether the feature is selected or not selected in the corresponding feature subset, respectively. For instance, a string of ten binary digits (i.e., a feature subset, a solution or a chromosome), say, 0011010001, means that features 3, 4, 6, and 10 are selected in the corresponding feature subset. This encoding process plays an essential role in the proposed method. The procedures of the proposed method for selecting representative features from the original feature set F are detailed as follows.

- Step1. Initialize a set (sub-population) CS of candidate feature subsets ($CS \subseteq S$). The population size of CS is denoted as ps .
- Step2. Randomly select (with replacement) ps parent solutions from CS by Roulette Wheel selection method [18]. Consequently, $ps/2$ pairs of parent solutions, denoted as set VS , are obtained.
- Step3. Generate ps offspring by $ps/2$ crossover operations [9] performed on the corresponding $ps/2$ pairs of parent solutions in VS . Here, a set of ps offspring solutions, denoted as BS_1 , is obtained.
- Step4. Perturb each offspring solution in BS_1 by a mutation operation [9] (The mutation probability is determined by a mutation rate, mr). Similarly, a set of ps perturbed offspring solutions, denoted as BS_2 , is obtained.
- Step5. Randomly select (with replacement) two solutions, denoted as b_1 and b_2 , from BS_2 . Consider that b_1 and b_2 have w different bits ($w \leq n$).
- Step6. Generate an “extended” two-level (levels 1 and 0) orthogonal array OA with respect to the above particular w bits (i.e., attributes, features or factors) of b_1 and b_2 . The level of feature i in OA will be replaced by the corresponding bit of b_1 if the original level is 0. Conversely, the level of feature i in OA will be replaced by the corresponding bit of b_2 if the original level is 1. Notably, the levels of the remainder $n-w$ features (i.e., attributes or factors) in the two-level orthogonal array OA are the same as the corresponding bits of b_1 and b_2 . In each experimental trial, j , in the new, extended two-level orthogonal array OA , levels 1 or 0 in each column i indicate that feature i is selected or not selected in the corresponding feature set S_j for pattern classification, respectively.
- Step7. For each feature set S_j , determine the average classification accuracy for training set T (denoted by $ACC(T, S_j)$) using the nearest neighbor classification rule [4][5] with the leave-one-out (LOO) cross-validation technique [3][19]. Here, $ACC(T, S_j)$ is considered an observation or objective function of experimental trial j in the new, extended two-level orthogonal array OA . This process, also named as fitness evaluation, is used to measure the goodness of each feature set or solution S_j .
- Step8. Calculate the corresponding signal-to-noise ratio (SNR) for each level (i.e., levels 1 and 0) of the particular w features according to observations from all experimental trials in the new, extended two-level orthogonal array OA .
- Step9. Generate a better solution t_best based on the results in the new, extended two-level orthogonal array OA . For all w bits (i.e., attributes, features or factors) in t_best , each bit is determined by value 1 if its corresponding SNR for level 1 is greater than that for level 0, and vice versa. Notably, the remainder $n-w$ bits (i.e., attributes, features or factors) of t_best are the same as those of b_1 and b_2 .
- Step10. Repeat Steps 5-9 until a set of $\lfloor ps * pc / 4 \rfloor$ better solutions (i.e., $\lfloor ps * pc / 4 \rfloor$ different t_bests), denoted as BS_3 , is obtained.
- Step11. Update CS by using better candidate feature subsets (i.e., solutions) in BS_2 and BS_3 .
- Step12. Repeat Steps 2-11 until a certain number of iterations have been completed. Consequently, the best feature subset, denoted as g_best in CS is utilized as the final feature subset for pattern classification.

As mentioned, in the proposed method, genetic algorithms are employed with randomness for “global search” of the entire search space (i.e., 2^n possible feature subsets). Various genetic operations, including crossover (Steps 2-3), mutation (Step 4), selection and replacement (Step 11) are performed to assist the search procedure in escaping from sub-optimal solutions [17].

Notably, the goodness of each feature set or solution in the above-mentioned procedures (including Steps 7, 11 and Roulette Wheel selection method [18] in Step 2) is evaluated using the nearest-neighbor classification rule [4][5] with the leave-one-out (LOO) cross-validation technique [3][19]. That is, classification accuracy with respect to training set T and a particular feature set or solution PS , denoted as $ACC(T, PS)$, can be determined. Leave-one-out cross-validation indicates that each instance in T is a test instance once and the other instances in T are considered corresponding training instances. Thus, the nearest-neighbor classification rule is applied m times according to m instances and n features in V . Next, average classification accuracy is calculated as a fitness evaluation function for analyzing the classification performance or goodness of the corresponding feature set PS .

During each iteration in the proposed approach, a set CS of candidate feature subsets is obtained. The Taguchi methods (Steps 5-10) can then help explore better feature subsets (or solutions), which are somewhat different from those in CS , for the next iteration. Conversely, the Taguchi methods are employed for “local search” of the entire search space (i.e., 2^n possible feature subsets). As stated, consider that two solutions, b_1 and b_2 , which are selected from BS_2 (a set of ps perturbed offspring solutions), have w different bits ($w \leq n$). The two-level orthogonal array is utilized for a fair, balanced and well-organized comparison of two levels for the w features (two levels indicate that the feature is selected or not selected for pattern classification) and interactions among features in a specific classification problem. The signal-to-noise ratio (SNR) is then employed to determine the robustness of the w features. Consequently, a superior candidate feature subset, which has high classification performance for the classification task, can be obtained by considering each feature with a specific level having a high SNR. Here, the larger-the-better characteristic is selected for calculating SNR as maximal classification accuracy is preferred in pattern classification. Here, feature i with an SNR for level 1 greater than that for level 0 indicates that the feature is will be selected in the candidate feature subset for pattern classification. Conversely, feature i is removed from the candidate feature subset when the corresponding SNR for level 0 is greater than that for level 1. As mentioned, if a particular target (e.g. process or product) has d different design factors, 2^d possible experimental trials will need to be considered in full factorial experimental design. Orthogonal arrays are principally utilized to decrease experimental efforts associated with these d design parameters. Finally, prior to the classification process, feature subset evaluation efforts can be significantly reduced.

4 Experimental Results

To demonstrate the performance of the proposed method, seventeen real datasets [1] were used for performance comparison. In the experiments, related parameters are detailed as follows. (a) Population size (ps) = 20. (b) Crossover rate (pc) = 1.0. (c) Mutation rate (mr) = 0.01. (d) Number of iterations in Step 12 = 30.

Table 2 represents the classification abilities or accuracies with respect to the above seventeen pattern classification tasks while the proposed GA-Taguchi-based feature selection method is performed or not. Of these classification domains, the average classification accuracies can be increased from 83.31% to 87.11% when the proposed method is applied. Experimental results indicate that the obtained feature subset is helpful for pattern classification. For example, the classification accuracy with respect to the Glass pattern classification problem can be significantly increased from 91.08% to 98.59% when the proposed method is performed.

Table 2. The classification abilities or accuracies with respect to the seventeen pattern classification tasks while the proposed GA-Taguchi-based feature selection method is performed or not

Classification task	The proposed method is not used for feature subset selection	The proposed method is used for feature subset selection
Australian	79.42	85.36
Balance-scale	78.24	79.20
Breast-cancer	95.61	97.07
Cmc	43.18	47.45
Diabetes	70.57	71.35
German	67.60	73.10
Glass	91.08	98.59
Heart	75.56	81.85
Ionosphere	86.89	94.87
Iris	95.33	95.33
Satellite	90.54	91.48
Segment	97.40	97.53
Sonar	85.70	96.15
Vehicle	69.74	75.06
Vowel	99.19	99.12
WDBC	95.25	97.89
Wine	94.94	99.44
Average	83.31	87.11

5 Conclusions

This work presents a novel GA-Taguchi-based feature selection method. Genetic Algorithms are utilized with randomness for “global search” of the entire search space. Various genetic operations are performed to assist the search procedure in escaping from sub-optimal solutions. In each iteration in the proposed method, the Taguchi methods are utilized for “local search” of the entire search space and thus can help explore better feature subsets for next iteration. As a result, feature subset evaluation efforts can be significantly reduced and a superior feature subset with high classification performance can be obtained. The proposed approach, with wrapper nature, yields superior performance and improves classification accuracy in pattern classification.

References

1. Blake, C.L., Merz, C.J.: UCI Repository of machine learning databases. University of California, Department of Information and a Computer Science, Irvine (1998), <http://www.ics.uci.edu/~mlearn/MLRepository.html>
2. Blum, A., Langley, P.: Selection of Relevant Features and Examples in Machine Learning. *Artificial Intelligence* 97, 245–272 (1997)
3. Cawley, G.C., Talbot, N.L.C.: Efficient Leave-one-out Cross-validation of Kernel Fisher Discriminant Classifiers. *Pattern Recognition* 36, 2585–2592 (2003)
4. Cover, T.M., Hart, P.E.: Nearest Neighbor Pattern Classification. *IEEE Trans. on Information Theory* 13, 21–27 (1967)
5. Dasarathy, B.V.: Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques. IEEE Computer Society Press, Los Alamitos (1990)
6. Dash, M., Liu, H.: Feature Selection for Classification. *Intelligent Data Analysis* 2, 232–256 (1997)
7. Doak, J.: An Evaluation of Feature Selection Methods and Their Application to Computer Security. Technical Report, Univ. of California at Davis, Dept. Computer Science (1992)
8. Duda, R.O., Hart, P.E.: Pattern Classification and Scene Analysis. John Wiley & Sons, Chichester (1973)
9. Goldberg, D.E.: Genetic Algorithms in Search, Optimization and Machine Learning. Addison Wesley, Reading (1989)
10. Hall, M.A.: Correlation-based Feature Subset Selection for Machine Learning. PhD Dissertation, University of Waikato (1998)
11. Holland, J.H.: Adaptation in natural and artificial systems. University of Michigan Press, Ann Arbor (1975)
12. Inza, I., Larrañaga, P., Sierra, B.: Feature Subset Selection by Bayesian Networks: a Comparison with Genetic and Sequential Algorithms. *International Journal of Approximate Reasoning* 27, 143–164 (2001)
13. John, G.H., Kohavi, R., Pfleger, K.: Irrelevant Feature and the Subset Selection Problem. In: Proc. 11th Int'l Conf. Machine Learning, pp. 121–129 (1994)
14. Kohavi, R., John, G.H.: Wrappers for Feature Subset Selection. *Artificial Intelligence* 97, 273–324 (1997)
15. Liu, H., Motoda, H.: Feature Selection for Knowledge Discovery and Data Mining. Kluwer Academic, Boston (1998)
16. Liu, H., Setiono, R.: A Probabilistic Approach to Feature Selection - A Filter Solution. In: Proc. of 13th International Conference on Machine Learning, pp. 319–327 (1996)
17. Liu, H., Yu, L.: Toward Integrating Feature Selection Algorithms for Classification and Clustering. *IEEE Trans. Knowl. Data Eng.* 17, 491–502 (2005)
18. Mitchell, M.: An Introduction to Genetic Algorithms. MIT Press, Cambridge (1992)
19. Stone, M.: Cross-validatory Choice and Assessment of Statistical Predictions. *Journal of the Royal Statistical Society B* 36, 111–147 (1974)
20. Taguchi, G., Chowdhury, S., Taguchi, S.: Robust Engineering. McGraw-Hill, New York (2000)
21. Wu, Y., Wu, A., Taguchi, G.: Taguchi Methods for Robust Design. ASME, New York (2000)

Nonnegative Matrix Factorization (NMF) Based Supervised Feature Selection and Adaptation

Paresh Chandra Barman and Soo-Young Lee*

Department of Bio and Brain Engineering, Brain Science Research Center (BSRC),
KAIST Daejeon-Korea
pcbarman@gmail.com, sylee@kaist.ac.kr

Abstract. We proposed a novel algorithm of supervised feature selection and adaptation for enhancing the classification accuracy of unsupervised Nonnegative Matrix Factorization (NMF) feature extraction algorithm. At first the algorithm extracts feature vectors for a given high dimensional data then reduce the feature dimension using mutual information based relevant feature selection and finally adapt the selected NMF features using the proposed Non-negative Supervised Feature Adaptation (NSFA) learning algorithm. The supervised feature selection and adaptation improve the classification performance which is fully confirmed by simulations with text-document classification problem. Moreover, the non-negativity constraint, of this algorithm, provides biologically plausible and meaningful feature.

Keywords: Nonnegative Matrix Factorization, Feature Adaptation, Feature extraction, Feature selection, Document classification.

1 Introduction

In machine learning systems feature extraction of the high-dimensional data such as text is an important step. The extracted feature vectors, individually or by combination with more than one represent the data as pattern or knowledge unit. In a text mining system, pattern or knowledge discovery is an important task. In some text classification works such as [1, 2] tried to reduce the dimension of original feature (word or term) set using the statistical measure of relevance i.e., mutual information among the terms and given category. These selected set of terms are used to represent the natural language text into numerical form by counting the term-document frequencies. However, the terms or words have properties such as polysemy, and synonymy, that does not make the words as optimal features. These problems motivated for text feature extraction such as clustering [3].

The unsupervised Nonnegative Matrix Factorization (NMF) algorithm [4] has been used to extract the semantic features from a document collection, where the algorithm provides two encoded factors with the approximation of $\mathbf{X} \approx \mathbf{BS}$. Here \mathbf{X} , \mathbf{B} and \mathbf{S} are

* Corresponding Author.

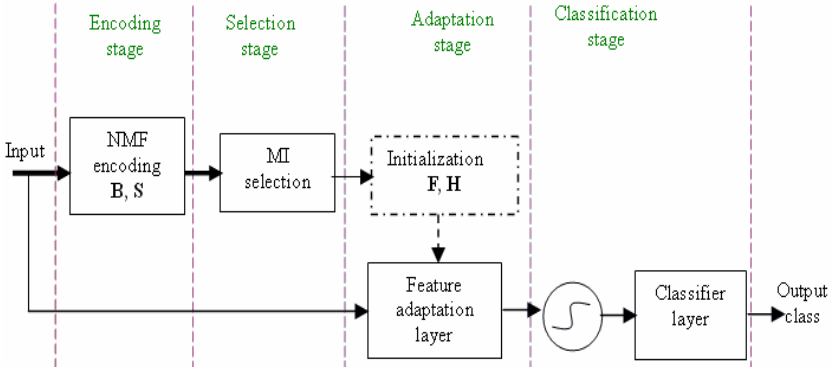


Fig. 1. NMF based feature selection and adaptation model for classification

the input data, the basis factor and the coefficient factor respectively, of which elements are all non-negative. The basis factor may be regarded as a set of feature vectors or patterns. The NMF and its extended versions [5, 6] become popular in many scientific and engineering applications including pattern clustering [7], blind source separation [5], text analysis [4], and music transcription [8].

Although NMF feature extraction process is based on the internal distribution of given data the algorithm is blind about the category and may not be optimal for classification. Therefore, one may select only relevant features for given category information and/or adapt the features for best classification performance. In general, the hidden layer of multilayer Perceptron (MLP) may be regarded as feature extractors with category information. The gradient-based learning for MLP has a tendency to stick with local minima, and its performance depends on good initialization of synaptic weights [9]. The first layer of MLP may be initialized by NMF based features, which can provide a good approximation of initial synaptic weights. With this initialization, to train the networks we modifying the general error back-propagation learning rule, and proposed a Non-negative Supervised Feature Adaptation (NSFA) algorithm.

2 Algorithm

We divide the whole classification process into four stages: unsupervised NMF feature extraction, feature selection based on Mutual Information (MI), non-negative supervised feature adaptation, and Single Layer Perceptron (SLP) classification as shown in Fig.1. Let us consider \mathbf{X} be a given input data matrix of dimension $N \times M$, of which each column represents a training sample with non-negative elements and $M \gg N$. The samples are normalized to have unit sum i.e., $x_{ij} = x_{ij} / \sum_i x_{ij}$. At the feature extraction stage one decomposes \mathbf{X} into a feature or basis matrix \mathbf{B} and coefficient matrix \mathbf{S} with the following update rules, which are based on generalized KL divergence [4] as

$$\begin{aligned}
 b_{ir} &\leftarrow b_{ir} \left(\left(\sum_j s_{rj} (X_{ij} / (\mathbf{BS})_{ij}) \right) / \sum_j s_{rj} \right) \\
 b_{ir} &= b_{ir} / \sum_i b_{ir} \\
 s_{rj} &\leftarrow s_{rj} \left(\left(\sum_i b_{ir} (X_{ij} / (\mathbf{BS})_{ij}) \right) / \sum_i b_{ir} \right) \\
 &\text{where } 1 \leq i \leq N, 1 \leq j \leq M, 1 \leq r \leq R
 \end{aligned} \tag{1}$$

here the number of NMF features is an empirical parameter R .

At the selection stage one may remove irrelevant features and keep only important features for the classification. For the selection criteria we chose the mutual information between the feature coefficient and category variable as

$$MI(\mathbf{s}_r, \mathbf{c}) = \sum_{k=1}^K \sum_{q=1}^Q \Pr(s_{rq}, c_k) \log_2 \frac{\Pr(s_{rq}, c_k)}{\Pr(s_{rq})\Pr(c_k)} \tag{2}$$

where $\Pr(a, b)$ is the joint probability of variables a and b , $\Pr(a)$ is the probability of a , K is the number of categories, \mathbf{c} is the category label vector, Q is the number of quantization levels for \mathbf{s}_r 's.

In general the features have mutual information among themselves, and the importance of a feature need to be evaluated by Information Gain (IG), which is defined as the additional information by adding a feature to an existing set of features. It requires calculation of higher dimensional joint probability, which requires huge data for accurate estimation. Since the cost function of our NMF feature extractor is a generalized version of KL divergence, the extracted features have small mutual information among themselves and the IG may be approximated by the simpler MI.

The features may be adapted to provide best classification performance. The feature adaptation and classification network can be considered as a two-layer feed-forward network, where the first layer is for feature adaptation and second layer for classification. With the first layer synaptic weight \mathbf{W} , hidden neuron activation \mathbf{H} is defined as $\mathbf{H} = \mathbf{W}\mathbf{X}$. While the initial values of \mathbf{W} are set to Moore-Penrose pseudo inverse of \mathbf{F} as $\mathbf{W}_0 = [\mathbf{F}]_{\epsilon}^{\dagger}$, here \mathbf{F} is the selected columns of the NMF feature matrix \mathbf{B} . The operator $[a]_{\epsilon}$ represents a half-wave transfer function of variable a , such as $a = \max(a, \epsilon)$, with ϵ is a very small positive number (e.g., $\epsilon = 10^{-16}$).

Let \mathbf{Y} be the input to the classifier layer and \mathbf{O} be the output of the classifier layer as

$$\begin{aligned}
 y_{rj} &= \frac{2}{1 + \exp(-\alpha h_{rj})} - 1 \quad \text{and} \\
 o_{kj} &= f(\mathbf{U}, \mathbf{y}) = \frac{1}{1 + \exp(-\beta (\sum_r u_{kr} y_{rj}))}
 \end{aligned} \tag{3}$$

here \mathbf{U} is the synaptic weight of the classifier layer. Let the error, $e_{kj} = t_{kj} - o_{kj}$, for a given target \mathbf{C} , then the synaptic weight \mathbf{U} can be updated by simple gradient rule as

$$u_{kj} \leftarrow u_{kj} + \Delta u_{kj} \quad (4)$$

and based on the error gradient the hidden activation \mathbf{H} can be modified as

$$h_{rj} \leftarrow [h_{rj} + \Delta h_{rj}]_e \quad (5)$$

where $\Delta h_{rj} = -\eta_1 \frac{\delta E_j}{\delta h_{rj}}$, $E_j = \frac{1}{2} \sum_{k=1}^K e_{kj}^2$, η 's are learning rate, and K is the total number of categories.

Now based on the NMF approximation we can estimate the hidden or coefficient matrix, with given input data \mathbf{X} and feature matrix \mathbf{W} as

$$\hat{\mathbf{H}} = \mathbf{W}\mathbf{X} \text{ or } \hat{h}_{rj} = \sum_i w_{ri} x_{ij} \quad (6)$$

Now we can minimize the Euclidean distance $\mathbf{D}(\mathbf{H}, \hat{\mathbf{H}}) = \|\mathbf{H} - \hat{\mathbf{H}}\|_2$ between updated and estimated \mathbf{H} with respect to \mathbf{W} as

$$\mathbf{W} \leftarrow \mathbf{W} + \eta \Delta \mathbf{W}; \text{ where } \Delta w_{ri} = -\frac{\delta D_j}{\delta w_{ri}} \quad (7)$$

Let us consider the instantaneous distance D_j for a certain training sample j , is defined as $D_j = \frac{1}{2} \sum_r d_{rj}^2$ and $d_{rj} = (h_{rj} - [wx]_{rj}) = (h_{rj} - \sum_i w_{ri} x_{ij})$, r is the number of features. The minimization of D_j with respect to \mathbf{W} is defined as

$$\Delta w_{ri} = -\frac{\delta D_j}{\delta d_{rj}} \frac{\delta d_{rj}}{\delta w_{ri}} = \sum_j h_{rj} x_{ji} - \sum_j [wx]_{rj} x_{ji} \quad (8)$$

By choosing an appropriate learning rate η , we can get the following multiplicative update rule for \mathbf{W} as

$$w_{ri} \leftarrow w_{ri} + \eta \Delta w_{ri}; \text{ with } \eta = \frac{w_{ri}}{\sum_j [wx]_{rj} x_{ji}} \quad (9)$$

$$w_{ri} \leftarrow w_{ri} \frac{\sum_j h_{rj} x_{ji}}{\sum_j [wx]_{rj} x_{ji}}$$

$$\text{In matrix form: } \mathbf{W} \leftarrow \mathbf{W} \cdot \left(\frac{\mathbf{H}\mathbf{X}^T}{[\mathbf{W}\mathbf{X}]\mathbf{X}^T} \right) \quad (10)$$

We start the feature adaptation process with the non-negativity constraint, such as $(\mathbf{W}, \mathbf{H}, \mathbf{X}) \geq 0$, and we get a multiplicative feature adaptation rule, as a result the adapted features would be non-negative and the algorithms will be consistent with the advantages of NMF algorithm [4] such as the features are meaningful only with additive nature and sparse or part-base. The NMF feature adaptation algorithm can be also defined as *MLP* learning by *NMF* initialization with nonnegative constraint (*MLP-NMFI-NC*).

3 Simulation

To observe the performance of our proposed algorithms, we selected 8137 documents from top eight document categories of Reuters21578 text database. We randomly generate four-fold training, validation and test data sets. In each fold data we use 50% documents for training, 25% for validation and the remaining 25% for test. In the pre-processing stage of representing the documents into term-document frequency matrix, we first use the stop-words elimination and category-based rare-word elimination [10], and select 500 terms with higher frequencies. The frequencies are normalized to have unit sum for each document.

To observe the classification performance of our proposed algorithm; we consider a baseline system, where the optimum extracted coefficient factor $\mathbf{H}=\mathbf{S}$ of the NMF algorithm with a predefined number of features (such as 2, 3, 4, 5, 6, 10 15, 25, 50 100, 150 300, and 500) is classified by training a Single Layer Perceptron (SLP) classifier, which is defined as *NMF-SLP* approach.

We verified the MI based feature selection approach without feature adaptation, in this case at first we extract very large number (500, same as the number of selected terms) of NMF features, then select some important feature subsets with different dimension (like 2, 3, 4, 5, 6, 10 15, 25, 50 100, 150 300, and 500), and finally test the classification performance with these selected ($\mathbf{H} \subseteq \mathbf{S}$) NMF coefficient factors by training a SLP classifier, which is defined as *MI-NMF-SLP* approach.

We also verified the feature adaptation approach without feature selection. In this case the feature extraction process is similar to NMF-SLP approach. For adaptation the network can be considered as MLP structure where the first layer synaptic weight is initialized by the pseudo-inverse of NMF optimized features and the learning algorithm is similar to MLP learning with nonnegative constraint which is defined as *MLP-NMFI-NC*.

Finally, MI based selected NMF features are adapted using the proposed algorithm which is defined as *MI-MLP-NMFI-NC*. Again here the feature extraction and selection processes are similar to *MI-NMF-SLP* approach. The first layer synaptic weight of the MLP network is initialized by the selected NMF features and adapted by the proposed multiplicative learning rule with nonnegative constraint.

4 Results

The connectivity of the terms corresponding to the feature vectors are non-negative i.e., is a terms is connected to a feature vector the synaptic weight has positive value

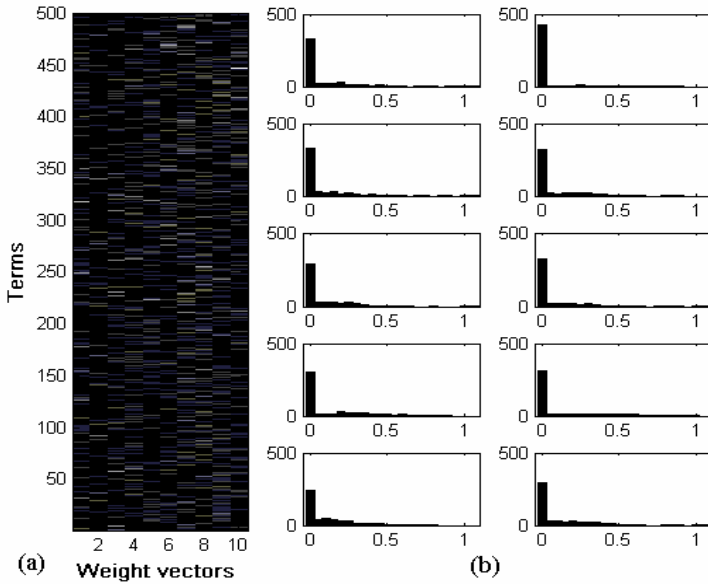


Fig. 2. (a) Synaptic connectivity of terms to feature vectors, (b) Synaptic Connection probability distributions of 10 feature vectors

otherwise zero, which is shown in Fig.2a here we have only presented the case when we select 10 NMF features using MI and then adapt those using NSFA algorithm. The features are sparse which has been shown in Fig.2b, where the histogram of 10 adapted feature vectors shows the number of terms (Y-axis in each figure) connected to a feature vectors with different synaptic weight strength, i.e., bin centre of the histogram (as shown in X-axis). This figure suggests that in each feature vector only few terms are connected (nonzero synaptic weight) and most are not connected (zero synaptic weight).

Based on mutual information measure, it has been observed that all the NMF extracted features are not important or relevant for classification. Few of that are more relevant to the categories such 50 most relevant NMF features based on MI score has been shown in Fig. 3a. The overall average classification performances for 4-fold data set with different classification approaches: *NMF-SLP*, *MI-NMF-SLP*, *MLP-NMFI-NC* and *MI-MLP-NMFI-NC* have been shown in Fig. 3b, corresponding to different number of features as indicated in the X-axis (in log scale). In general the unsupervised NMF algorithm optimizes and extracts almost independent and sparse features, and the optimum feature dimension R is empirical. In our observation, the classification performance of NMF-SLP approach is proportionally dependent on the feature dimension. The MI-NMF-SLP approach also shows the similar trend of performance dependency as NMF-SLP approach. In MI-NMF-SLP case, we select a few features from a large number of independent and optimum NMF feature set, due to the selection the features become suboptimal so the performance is worst. The MLP-NMFI and MI-MLP-NMFI-NC algorithms are performed better with small number of features owing to the adaptation based on given category information. Although the

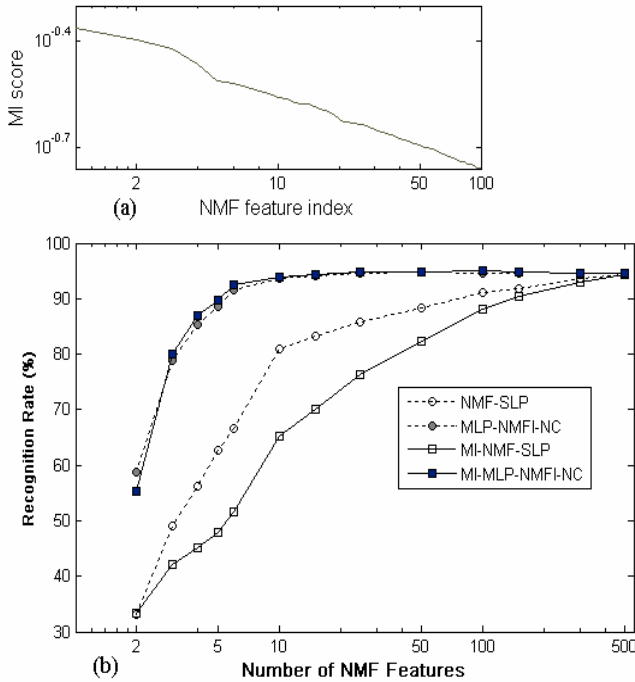


Fig. 3. (a) Mutual information score for top 100 NMF features, (b) Classification performance for different algorithms

performance of MLP-NMFI-NC is almost similar to MI-MLP-NMFI-NC but in the second approach is more effective. In first case we need to extract NMF features with different predefined number of basis while in MI-MLP-NMFI-NC case we just extract once NMF features with large number of basis then the relevant features are selected based on MI score. With small number of adapted features (about 25 to 50) are just enough to represent the data properly. We can consider these adapted features as the representative patterns or knowledge of the given data or document collection.

5 Conclusion

In this work we have presented a supervised algorithm for effective feature extraction, dimension reduction and adaptation. This algorithm will extend the application area of unsupervised NMF feature extraction algorithm into the supervised classification problems. The non-negativity constraint of this algorithm represents the data by sparse patterns or feature vectors that are understandable in text mining application. The better performance with small number of features can be useful to make a faster classifier system. We are going to implement our idea of NMF feature adaptation using standard error back propagation algorithm, which will be helpful to reduce the risk of local minima in MLP training.

Acknowledgement

This work was supported as the Brain Neuroinformatics Research Program by Korean Ministry of Commerce, Industry and Energy.

References

1. Makrehchi, M.M., Kamel, M.S.: Text classification using small number of features. In: Perner, P., Imiya, A. (eds.) *MLDM 2005*. LNCS (LNAI), vol. 3587, pp. 580–589. Springer, Heidelberg (2005)
2. Yang, Y., Pedersen, J.O.: A comparative study on feature selection in text categorization. In: *Proc. of 14th ICML*, pp. 412–420 (1997)
3. Slonim, N., Tishby, N.: The power of word clusters for text classification. In: *Proc. ECIR* (2001)
4. Lee, D.D., Seung, H.S.: Learning of the parts of objects by nonnegative matrix factorization. *Nature* 401, 788–791 (1999)
5. Cichocki, A., Zdunek, R.: Multilayer Nonnegative Matrix Factorization. *Electronics Letters* 42, 947–948 (2006)
6. Hoyer, P.O.: Non-negative Matrix Factorization with Sparseness Constraints. *Journal of Machine Learning Research* 5, 1457–1469 (2004)
7. Shahnaz, F., Berry, M.W., Paul, P., Plemmons, R.J.: Document clustering using nonnegative matrix factorization. *Information Processing and Management* 42, 373–386 (2006)
8. Smaragdis, P., Brown, J.C.: Non-Negative Matrix Factorization for Polyphonic Music Transcription. In: *Proc. WASPAA 2003*. IEEE press, Los Alamitos (2003)
9. Haykin, S.: *Neural Networks a comprehensive foundation*, 2nd edn. Prentice Hall, Englewood Cliffs (1999)
10. Barman, P.C., Lee, S.Y.: Document Classification with Unsupervised Nonnegative Matrix Factorization and Supervised Perceptron Learning. In: *Proc. ICIA 2007*, pp. 182–186. IEEE Press, Korea (2007)

Automatic Order of Data Points in RE Using Neural Networks

Xueming He^{1,2,*}, Chenggang Li¹, Yujin Hu¹, Rong Zhang³, Simon X. Yang⁴,
and Gauri S. Mittal⁴

¹ School of Mechanical Science and Engineering, Huazhong University of Science and
Technology, Wuhan 4300743

hxxuem2003@163.com

² School of Mechanical Engineering, Jiangnan University, Wuxi, Jiangsu, China 214122

³ School of Science, Jiangnan University, Wuxi, Jiangsu, China 214122

⁴ School of Engineering, University of Guelph, Guelph, Ont., Canada N1G2W1
Tel.: 86-510-591-0531

Abstract. In this paper, a neural network-based algorithm is proposed to explore the order of the measured data points in surface fitting. In computer-aided design, the ordered points serve as the input to fit smooth surfaces so that a reverse engineering (i.e. RE) system can be established for 3D sculptured surface design. The geometry feature recognition capability of back-propagation neural networks is explored in this paper. Scan or measuring number and 3D coordinates are used as the inputs of the proposed neural networks to determine the curve to which each data point belongs and the order number of data point in the same curve. In the segmentation process, the neural network output is segment number; while the segment number and sequence number in the same curve are the outputs when sequencing the points in the same curve. After evaluating a large number of trials with various neural network architectures, two optimal models are selected for segmentation and sequence. The proposed model can easily adapt for new data from another sequence for surface fitting. In comparison to Lin et al.'s (1998) method, the proposed algorithm neither needs to calculate the angle formed by each point and its two previous ones nor causes any chaotic phenomenon.

Keywords: automatic order, segment and sequence, neural networks, reverse engineering, surface fitting.

1 Introduction

Computer aided design and manufacturing (CAD/CAM) play an important role in present product development. Many existing physical parts such as car bodies, ship hulls, propellers and shoe insoles need to be reconstructed when neither their original drawings nor CAD models are available. The process of reproducing such existing parts is called as reverse engineering. In this situation, designers must employ either a

* Corresponding author.

laser scanner or a coordinate measuring machine to measure existing parts, and then use the measured data to reconstruct a CAD model. After the CAD model is established, it will then be improved by successive manufacturing-related actions.

From measuring point of view, it is difficult to fetch surface information through CMM and the massive point data obtained can barely be directly processed (Yau et al. 1993). An automated inspection planning system is developed to measure the point data of a surface profile (Yau and Menq 1993). Based on least-square theory, point data are fit into a nonuniform rational B-spline surface (NURBS) and a solid model respectively (Sarkar and Menq 1991, Chivate and Jablolkow 1993). The geometric continuity of B-spline or Bezier surface patches in reverse engineering is also an important problem (Du and Schmitt 1990, and Milroy et al. 1995). There is a review on data acquisition techniques, characterization of geometric models and related surface representations, segmentation and fitting techniques in reverse engineering (Tamas Varady et al. 1997).

For sequencing the measured point data and increasing the accuracy of fitting surface, four algorithms were proposed to categorize and sequence the continuous, supplementary and chaotic point data and revise the sequence of points (Lin et al 1998). Through sampling, regressing and filtering, the points were regenerated with designer interaction in the format that they meet the requirements of fitting into a B-spline curve with good shape and high quality (Tai et al 2000). Using the normal values of points, a large amount of unordered sets of point data were handled and constructed the final 3D-girds by the octree-based 3D-grid method (Woo et al 2002).

Conventional computation methods process the point data sequentially and logically, and the description of given variables are obtained from a series of instructions to solve a problem. In comparison to them, artificial neural networks (NNs) can be used to solve a wide variety of problems in science and engineering, especially for the fields where the conventional modeling methods fail. Particularly, NNs do not need to execute programmed instructions but respond in parallel to the input patterns (Ward Systems Group 1998). Well-trained NNs can be used as a model for a specific application, which is a data-processing system inspired by a neural system. They can capture a lot of relationships or discover regularities existing in a set of input patterns that are difficult to describe adequately by conventional approaches.

Feature extraction has been widely investigated for many years. A NN is employed to recognize features from boundary representations solid models of parts described by the adjacency matrix, which contains input patterns for the network (Prabhakar and Henderson 1992). However, in their work, no training step was addressed in the repeated presentations of training. To reconstruct a damaged or worn freeform surface, a series of points from a mathematically known surface were applied to train a NN (Gu and Yan 1995). A set of heuristics was used to break compound features into simple ones using a NN (Nezis and Vosniakos 1997). Freeform surfaces from Bezier patches were reconstructed by simultaneously updating networks that corresponded to the separate patches (Knopf et al. 2001). A back propagation network was used to segment surface primitives of parts by computing Gaussian and mean curvatures of the surfaces (Alrashdan et al. 2000). The recommended number of measuring points for a rule surface was explored by applying a NN (Lin et al. 2000). A neural network self-organizing map (SOM) method was employed to create a 3D parametric grid and reconstruct a B-Spline surface (J. Barhak and A. Fisher 2001, 2002). Considering the

input parameterization did not reflect the nature of a tensor product B-Spline surface, a simple surface with a few control points (cubic Bézier patch) was built and then a reference surface was generated through gradually increasing the smoothness or the number of control points (V. Weiss et al 2002). After computing the best fit least-squares surfaces with adaptively optimized smoothness weights, a smooth surface within the given tolerances was found. A meshless parameterization was proposed to parameterize and triangulate unorganized point sets digitized from single patch (Michael S. Floater and Martin Reimers 2001). By solving a sparse linear system the points were mapped into a planar parameter domain and by making a standard triangulation of the parameter points a corresponding triangulation of the original data set were obtained. The chosen data points are used as control points to construct initial NURBS surfaces and then all the measured data points are appended as target points to modify these initial surfaces using minimization of deviation under boundary condition constraints (Zhongwei Yin 2004). Assuming the normal at the unorganized sample points known, smooth surfaces of arbitrary topology are reconstructed using natural neighbor interpolation (Jean-Daniel Boissonnat, Frédéric Cazals 2002).

2 Research Methodology

In reverse engineering, a designer firstly places the existing product in either a CMM or a laser scanner for surface configuration measurement. However, before the point data are processed by CAD system, the point data must be divided into several segments sorted to understand the start and end points of each curve, and then fit the data into curves and transformed the curves into surfaces. Currently, this procedure is conducted manually, but the precision can not be guaranteed.

On the other hand, when a contact device is used to measure the surface configuration of a product, the measured point data are sometimes not too dense to develop into satisfactory curves, surfaces and shape. More points are then needed to be measured on the portions where point data were not previously taken. The point data, acquired from the second round of measurements, have to be attached at the end of the original data. Consequently, before we utilize CAD software to build the surface model, the positions of these points added in the point data file must to be recognized and located precisely. Though relatively troublesome, this process proved an ineluctability step in the reverse engineering. Due to the versatile geometry patterns of different surface regions on the product body, the measurement and manual point-ordering procedure become even more complicated and troublesome. So Lin et al. (1998) have employed four continuous algorithms for cross-checking the point data with distances and angles to categorize and index. The procedure of dealing with the point data proposed by Lin et al. (1998) is very complicated and often causes chaotic and wrong order. Furthermore, for different shape geometry or a little complicated surface, their method must adjust and modify the angle threshold to meet the new demand. It is thus the objective of this paper to propose a viable, neural network-based solution to automatically segment and sequence the point data acquired by CMM, so as to simplify and fasten the processes for the measured point data usage in reverse engineering.

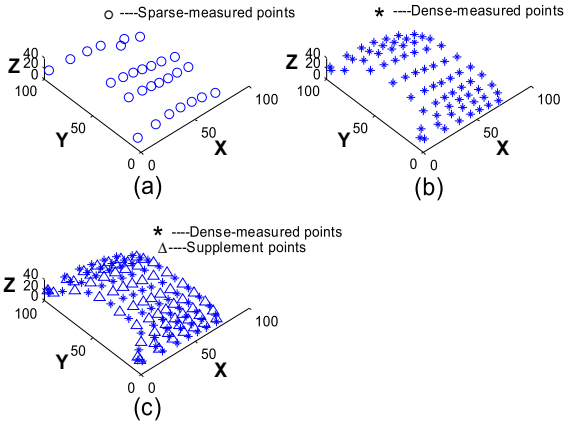


Fig. 1. Examples of measured point data of a toy-car engine-hood. (a) sparse; (b) dense; (c) supplementary appending.

here we must judge whether continuous points are located on the same curve. Since the geometry of surfaces varies, it is more difficult for us to use general algorithms let alone manual method to separate and locate such many measured point data. The first step of proposed method can easily and accurately solve the problem based on learning and testing the situation of Fig. 1(a) using NN architecture.

Fig. 1(c) show the result of supplementing points that are inserted at cleaves of the original measurement points to increase the accuracy of surface fitting. The 63 points obtained at the second round measurement are attached to the end of the original data file, while first 70 points are duplicated with the same sequencing method as the former ones. Now, it is very important to separate the second round point data into the previous curves and reorder those point data in correct sequence in the same curve. The second step of proposed method would be designed to solve accurately the problem based on learning and testing the result of the first step of proposed method using NN architecture.

3 Neural Network Design

To get the best segmentation and sequence by a NN, 16 neural network architectures shown in Fig. 2 were evaluated. The layers, the hidden neurons, scale and activation functions, learning rate, momentum and initial weights were varied in NNs. The algorithms used for NN training consisted of a back-propagation (BP) shown in Fig. 2(a)–(l), a general regression (GR) illustrated in Fig. 2(m), a multi-layer group method of data handling (GMDH) given in Fig. 2(n), an unsupervised Kohonen net and a probabilistic net shown in Fig. 2(o) and Fig. 2(p). In the segmentation procedure of car-toy engine-hood, there were just 28 datasets available to identify manually the start point and the end point of each curve. Five datasets of the total were randomly selected as testing sets and another 2 datasets as production sets. The remaining datasets were used for training the NN. While in the sequence procedure of the above product, there

In consideration of NN pattern recognition capability, the point data in a 3D space may be obtained from measurement at very sparse manner then a little dense and finally at very dense manner to fit the accurate surface gradually. Fig. 1 shows point data measured from the engine-hood of a toy-car model. Such sparse data (Fig. 1(a)) can directly manually identify the start and the end point of each curve. Fig. 1(b) show dense measured point data from the same model. As a result,

were more datasets than 70. If more datasets were selected as testing sets, production sets and training sets of the NN, the better precision should be obtained. Several different variations of BP networks were used.

A variety of NN architectures and the learning strategies evaluated in the NN model design were first adopted one by one. Then various scale and activation functions evaluated for the presented NN architecture were selected. After that, parameters such as the learning rate, momentum and initial weights, layers and hidden neurons were addressed. Finally, through learning and testing, optimal NN algorithms for two-step sequencing were obtained.

3.1 Neural Network (NN) Architecture and Learning Strategy

To search for optimal NNs architecture for automatic segmenting and sequencing 3D data points for surface fitting, 5 types of NN were evaluated, including BP, GR, GMDH, K, and P nets.

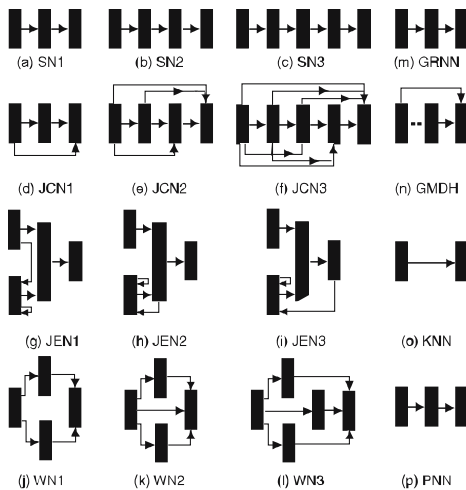


Fig. 2. Architecture of NN for the automatic ordering of data points

Jordan-Elman nets [JENs Fig. 2(g)–(i)] are recurrent networks with dampened feedback and there is a reverse connection from the hidden layer, from the input and output slab. Ward Systems Group (Frederick, MD) invented Ward nets [WNNs Fig. 2(j)–(l)]. The output layer, due to a direct link with the input layer, is able to spot the linearity that may be present in the input data, along with the features.

General regression neural network [GRNN Fig. 2(m)] is a three-layer network that learns in only one epoch, needs one hidden neuron for every pattern and works well on problems with sparse data. There was no training parameter, but a smoothing factor was required.

Back-propagation neural network (BPNN) is the most commonly used NN. There are mainly four types of BPNN: standard, jump connection, Jordan-Elman and Ward. For standard nets [SNs Fig. 2(a)–(c)], each layer was just connected to its immediately previous layer and the number of hidden layers varied from one to three for the best NN architecture. For jump connection nets [JCNs Fig. 2(d)–(f)], each layer was connected to every other layer. This architecture enabled every layer to view the features detected in all the previous layers as opposed to just the previous layer. Jordan-Elman nets [JENs Fig. 2(g)–(i)] are recurrent networks with dampened feedback and there is a reverse connection from the hidden layer, from the input and output slab. Ward Systems Group (Frederick, MD) invented Ward nets [WNNs Fig. 2(j)–(l)]. The output layer, due to a direct link with the input layer, is able to spot the linearity that may be present in the input data, along with the features.

Group method of data handling or polynomial network [GMDH Fig. 2(n)] is implemented with non-polynomial terms created by using linear and non-linear regression in the links.

Kohonen net [KNN Fig. 2(o)] is an unsupervised learning net. No desired outputs are provided. The net looked for similarities inherent in the input data and clusters them based on their similarity.

Probabilistic Neural Network [PNN Fig. 2(p)] has been implemented as a three-layer supervised network that classifies patterns. This network couldn't be used for problems requiring continuous values for the output. It could only classify inputs into a specified number of categories.

3.2 Scale and Activation Functions

To observe their effectiveness, both linear and non-linear scaling methods were evaluated in the NN model. Two non-linear scaling functions are logistic and tanh that scaled data to (0, 1) and (-1, 1), respectively.

There are eight activation functions in the NN model:

Linear: $G(u) = u$.

Non-linear: Logistic: $G(u) = \frac{1}{1 + \exp(-u)}$ Symmetric-logistic: $G(u) = \frac{2}{1 + \exp(-u)} - 1$

Gaussian: $G(u) = \exp(-u^2)$ Gaussian-complement: $G(u) = 1 - \exp(-u^2)$

Hyperbolic Tangent Tanh: $G(u) = \tanh(u)$ Tanh15: $G(u) = \tanh(1.5u)$

Sine: $G(u) = \sin(u)$

3.3 Learning Rate, Momentum and Initial Weights

Weight changes in back propagation were proportional to the negative gradient of the error. The magnitude change depends on the appropriate choice of the learning rate (η) (Anderson, 1995). In the learning of a NN, the right value of η was between 0.1 and 0.9.

The momentum coefficient (α) determined the proportion of the last weight change that was added into the new weight change. Values for the α could be obtained adaptively as η .

Training was generally commenced with randomly chosen initial weight values (ω). Since larger weight magnitude might drive the first hidden nodes to saturation, requiring large amounts of training time to emerge from the saturated state, 0.3 was selected as ω for all neurons in the NN.

3.4 Number of Hidden Neurons

Many important issues, such as determining how many training samples were required for successful learning, and how large a NN was required for a specific task, were solved in practice by trial and error. For a three-layer NN, the minimum hidden neurons could be computed according to the following formula (NeuroShell 2, Release 4.0 help file):

$$N_{hidden} \geq \frac{N_{input} + N_{output}}{2} + \sqrt{N_{datasets}},$$

Thus, in the proposed segmentation NN model, the minimum number of hidden neurons for three-, four- and five-layer networks was 8, 4 and 3; while in the proposed sequence NN model, these were 11, 6 and 4.

3.5 Optimal Artificial Neural Network

Firstly, working with 362 models in BP, Kohonen and Probabilistic networks, the best two models were selected based on feature recognition accuracy. Secondly, working with the two selected models, the optimum scale and activation functions were found. Lastly, using GRNN and GMDH, the optimum learning rate, momentum and initial weights of the NN were also found. The evaluating method for selecting optimal NN was based on the minimization of deviations between predicted and observed values. The statistical parameters S_r , S_d and R^2 were used to evaluate the performance of each NN (NeuroShell 2, Release 4.0 help file).

$$S_r = \sum(v - \tilde{v})^2, S_d = \sum(v - \bar{v})^2, R^2 = 1 - S_r/S_d,$$

At the same time, the following errors were calculated: the mean squared error (E) was the mean of the square of the actual minus the predicted values; the mean absolute error (\mathcal{E}) is the mean of the absolute values of the actual minus the predicted; and the mean relative error (\mathcal{E}') is the mean of absolute error divided by the actual value. The standard deviation δ, δ' of \mathcal{E} and \mathcal{E}' , and % data within 5% error were also calculated.

After testing and evaluating a large number of NNs, the optimal NNs for two-step automatic sequence of point data are the four-layer WN3.

4 Implementation

As sculpture surface was very complicated, it was necessary for surface fitting to scan enough points. It involved in categorizing these points into different segments, and then sequencing the segmented points into continuous points. The proposed method employs two NN models to do two operations.

4.1 Segmentation

This step aimed to train and test many NNs on the relatively simple and sparse point data from the sculpture surface to find an optimal NN architecture for segmenting the

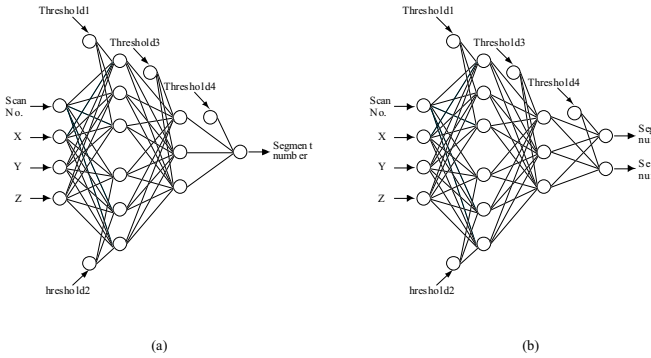


Fig. 3. Configuration of a NN for the automatic sequencing of data points

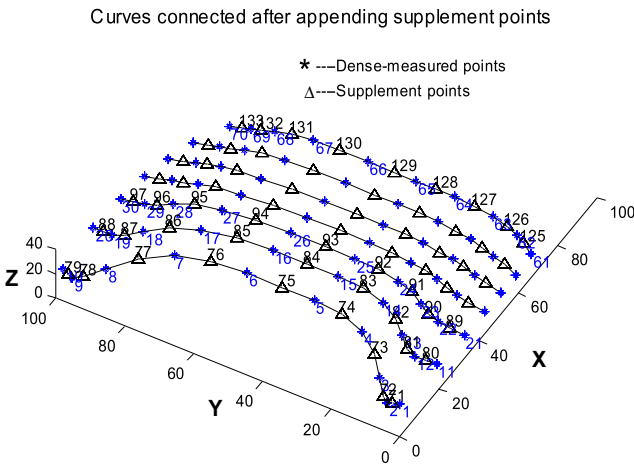


Fig. 4. Automatic sequencing of supplemental plus dense measured data points using a NN

Gaussian-complement respectively. The inputs of the NN were scan number and 3D coordinates, and the output was the curve or segment number. It was used for the dense measured point data of Fig. 1 to segment and categorize these dense points into several curves illustrated in Fig. 4. The first 10 points (marked by blue star) with the actual network output values from 1.003 to 1.249 closer to 1.0 represent their locating on the same curve i.e. curve 1. The second 10 points with the actual network output values from 1.944 to 2.216 closer to 2.0 consist of another curve i.e. curve 2. The other points construct the other curves. Thus, all points are separated into 7 curves and there are 10 points in each curve. Shown in Fig. 4, the first curve actually is made up of the most previous 10 points i.e. point 1, 2, 3... 9 and 10; the continuous 10 points consist of the second curve and so on.

point data. The optimal NN was then trained with both learning rate (η) and momentum (α) equal 0.1, and the initial weights ω equaled to 0.3 for all neurons in the NN. The adjustment of the weights was made to reduce the error between actual and predicted values after all the training patterns were presented to the network. The trained NN was then applied on the dense continuous point data from the same surface. Final optimal NN architecture for segmentation of point data was four-layer WN3 that contains 3 neurons for the second, third and fourth slabs (Fig. 3a). The scaling function was linear [0,1]; activation functions for the second, third, fourth and fifth slabs were Tanh, Sine, and

Sequencing of the supplementary and previous points of Fig. 1(c) is illustrated in Fig 4 and Table 1. The first example point 1 (scan no.) has the network output values of 1.000 (segment no.) and 1.057 (sequence no.), the second example point 71 has the output values of 1.000 and 1.378, and the third example point 2 has the output values of 1.006 and 1.629, indicate that points 1, 71 and 2 occupy the first curve and are continuous. According to the predications of NN, the other points on the first curve are 72, 3, 73 ... 9, 79, and 10. Another 19 points are on the second curve and so on. Thus, point 1 is smoothly connected 71, 2, 72... and 10 on the first curve; and points 11, 80, 12, 81... and 20 construct the second curve. Comparing Table 1 with Fig 4 indicates that the predicted segment and sequence numbers of the curves for all points accord fully with the actual points' positions. Fig. 5(a) is a surface drawn from the dense-measured points, while Fig. 5(b) is a more accurate surface fit after appending the supplemental points.

The prediction error analysis is given in Table 2. The mean absolute error for two-step ordering varied from 2.5% (for segment number) to 5.9% (for sequence number). The mean relative error also varied from 0.2% to 2.2%. From prediction error analysis and comparison of figures of predicted plan of point data to their actual locations, it is clear that the proposed NN model was valid for automatic sequence of point data for reverse engineering.

Table 2. Prediction errors analysis for the optimum NN

Statistical Parameters	Predication for sequencing of point data using WN3		
	Segmentation step		Sequence step
Patterns processed	Training and testing: 28 Production: 70		Training and testing: 70 Production: 133
	Output	Segment No.	Segment No. Sequence No.
Coefficient of determination R^2	0.9997	0.999	0.9992
Mean squared error E	0.001	0.004	0.007
Mean absolute error \mathcal{E}	0.025	0.041	0.059
Standard deviation σ of \mathcal{E}	0.032	0.063	0.084
Mean relative error \mathcal{E}'	0.008	0.002	0.022
Standard deviation σ' of \mathcal{E}'	0.075	0.065	0.126
Percent within 5% error	92.857	92.857	94.286
Percent within 5% to 10% error	7.143	4.286	4.286
Percent within 10% to 20% error	0	2.857	1.429

5 Conclusions

A novel sequencing neural network methodology for a set of scanned point data has been developed. The method could either automatically add supplementary data to the original, or re-establish the data points order without angle computation and chaotic phenomena. The neural network approach is a flexible reverse engineering methodology that could be easily trained to handle data points. The trained model has successfully

recognized geometric features such as curve and order of data points. The typical geometric characteristics of input patterns were automatically derived from the scanned points. Then they were presented to the neural network model to recognize appropriate geometric features. The ordered data points were further used for non-uniform rational B-spline surface model. The operator could observe how non-uniform rational B-spline surfaces differ from the actual, and decide whether denser data points were required.

The optimal neural network for segmentation and sequence of point data were four-layered Ward nets, whose activation functions were Tanh, Sine, and Gaussian-complement Logistic for segmentation procedure and Symmetric-logistic, and Gaussian-complement for sequencing procedure. The mean absolute error varied from 2.5 to 5.9% for predictions for two-step sequencing process of point data with an average error of 4%. The data within 5% error were more than 94.3% in sequencing process. Reasonable results were obtained from well-trained neural network as an alternative tool for surface fitting.

Acknowledgements

This work has been supported by the National Natural Science Foundation of China No.50575082.

References

- [1] Yau, H.T., Haque, S., Menq, C.H.: Reverse engineering in the design of engine intake and exhaust ports. In: Proceedings of the Symposium on Computer-Controlled Machines for Manufacturing, SAME Winter Annual Meeting, New Orleans, LA, USA (1993)
- [2] Yau, H.T., Menq, C.H.: Computer-aided coordinate metrology. In: 13th Annual ASME International Computers in Engineering Conference and Exposition, San Diego, CA, USA (August 1993)
- [3] Sarkar, B., Menq, C.H.: Smooth-surface approximation and reverse engineering. *Computer-Aided Design* 23(9), 623–628 (1991)
- [4] Chivate, T.N., Jablowski, A.G.: Solid-model generation from measured point data. *Computer-Aided Design* 25(9), 587–600 (1993)
- [5] Milroy, M.J., Bradley, C., Vickers, G.W., Weir, D.J.: G^1 continuity of B-spline surface patches in reverse engineering. *Computer-Aided Design* 27(6), 471–478 (1995)
- [6] Du, W.H., Schmitt, F.J.M.: On the G^1 continuity of piecewise Bezier surfaces: a review with new results. *Computer-Aided Design* 22(9), 556–573 (1990)
- [7] Varady, T., Martin, R.R., Cox, J.: Reverse engineering of geometric models—an introduction. *Computer-Aided Design* 29(4), 255–268 (1997)
- [8] Lin, A.C., Lin, S.-Y., Fang, T.-H.: Automated sequence arrangement of 3D point data for surface fitting in reverse engineering. *Computer in Industry* 35, 149–173 (1998)
- [9] Tai, C.-C., Huang, M.-C.: The processing of data points basing on design intent in reverse engineering. *International Journal of Machine Tools & Manufacture* 40, 1913–1927 (2000)
- [10] Woo, H., Kang, E., Wang, S., Lee, K.H.: A new segmentation method for point cloud data. *International Journal of Machine Tools & Manufacture* 42, 167–178 (2002)

- [11] Prabhakar, S., Henderson, M.R.: Automatic form-feature recognition using neural-network-based techniques on boundary representations of solid models. *Computer-Aided Design* 24(7), 381–393 (1992)
- [12] Gu, P., Yan, X.: Neural network approach to the reconstruction of freeform surfaces for reverse engineering. *Computer-Aided Design* 27(1), 59–64 (1995)
- [13] Nezis, K., Vosniakos, G.: Recognising 2.5D shape features using a neural network and heuristics. *Computer-Aided Design* 29(7), 523–539 (1997)
- [14] Ward Systems Group, Inc. (Frederick, MD), *NeuroShell 2 Help* (1998)
- [15] Barhak, J., Fisher, A.: Parameterization and reconstruction from 3D Scattered points based on neural network and PDE techniques. *IEEE Transactions on visualization and computer graphics* 7(1), 1–16 (2001)
- [16] Barhak, J., Fisher, A.: Adaptive reconstruction of freeform objects with 3D SOM neural network grids. *Computers & Graphics* 26, 745–751 (2002)
- [17] Weiss, V., Andor, L., Renner, G., Várady, T.: Advanced surface fitting techniques. *Computer Aided Geometric Design* 19, 19–42 (2002)
- [18] Floater, M.S., Reimers, M.: Meshless parameterization and surface reconstruction. *Computer Aided Geometric Design* 18, 77–92 (2001)
- [19] Yin, Z.: Reverse engineering of a NURBS surface from digitized points subject to boundary conditions. *Computers & Graphics* 28, 207–212 (2004)
- [20] Boissonnat, J.-D., Cazals, F.: Smooth surface reconstruction via natural neighbor interpolation of distance functions. *Computational Geometry* 22, 185–203 (2002)
- [21] Alrashdan, A., Motavalli, S., Fallahi, B.: Automatic segmentation of digitized data for reverse engineering application. *IIE Transactions* 32, 59–69 (2000)

Orthogonal Nonnegative Matrix Factorization: Multiplicative Updates on Stiefel Manifolds

Jiho Yoo and Seungjin Choi

Department of Computer Science
Pohang University of Science and Technology
San 31 Hyoja-dong, Nam-gu, Pohang 790-784, Korea
{zentasis,seungjin}@postech.ac.kr

Abstract. Nonnegative matrix factorization (NMF) is a popular method for multivariate analysis of nonnegative data, the goal of which is decompose a data matrix into a product of two factor matrices with all entries in factor matrices restricted to be nonnegative. NMF was shown to be useful in a task of clustering (especially document clustering). In this paper we present an algorithm for orthogonal nonnegative matrix factorization, where an orthogonality constraint is imposed on the nonnegative decomposition of a term-document matrix. We develop multiplicative updates directly from true gradient on Stiefel manifold, whereas existing algorithms consider additive orthogonality constraints. Experiments on several different document data sets show our orthogonal NMF algorithms perform better in a task of clustering, compared to the standard NMF and an existing orthogonal NMF.

1 Introduction

Nonnegative matrix factorization (NMF) is a multivariate analysis method which is proven to be useful in learning a faithful representation of nonnegative data such as images, spectrograms, and documents [1]. NMF seeks a decomposition of a nonnegative data matrix into a product of basis and encoding matrices with all of these matrices restricted to have only nonnegative elements. NMF allows only non-subtractive combinations of nonnegative basis vectors to approximate the original nonnegative data, possibly providing a parts-based representation [1]. Incorporating extra constraints such as locality and orthogonality was shown to improve the decomposition, identifying better local features or providing more sparse representation [2]. Orthogonality constraints were imposed on NMF [3], where nice clustering interpretation was studied in the framework of NMF.

A prominent application of NMF is in document clustering [4,5], where a decomposition of a term-document matrix was considered. In this paper we consider *orthogonal NMF* and its application to document clustering, where an orthogonality constraint is imposed on the nonnegative decomposition of a term-document matrix. We develop new multiplicative updates for orthogonal NMF, which are directly derived from true gradient on Stiefel manifold, while existing algorithms consider additive orthogonality constraints. Experiments on several

different document data sets show our orthogonal NMF algorithms perform better in a task of clustering, compared to the standard NMF and an existing orthogonal NMF.

2 NMF for Document Clustering

In the vector-space model of text data, each document is represented by an m -dimensional vector $\mathbf{x}_t \in \mathbb{R}^m$, where m is the number of terms in the dictionary. Given N documents, we construct a term-document matrix $\mathbf{X} \in \mathbb{R}^{m \times N}$ where X_{ij} corresponds to the significance of term t_i in document d_j that is calculated by

$$X_{ij} = \text{TF}_{ij} \log \left(\frac{N}{\text{DF}_i} \right),$$

where TF_{ij} denotes the frequency of term t_i in document d_j and DF_i represents the number of documents containing term t_i . Elements X_{ij} are always nonnegative and equal zero only when corresponding terms do not appear in the document.

NMF seeks a decomposition of $\mathbf{X} \in \mathbb{R}^{m \times N}$ that is of the form

$$\mathbf{X} \approx \mathbf{U}\mathbf{V}^\top, \quad (1)$$

where $\mathbf{U} \in \mathbb{R}^{m \times K}$ and $\mathbf{V} \in \mathbb{R}^{N \times K}$ are restricted to be nonnegative matrices as well and K corresponds to the number of clusters when NMF is used for clustering. Matrices \mathbf{U} and \mathbf{V} , in general, are interpreted as follows.

- When columns in \mathbf{X} are treated as data points in m -dimensional space, columns in \mathbf{U} are considered as *basis vectors* (or *factor loadings*) and each row in \mathbf{V} is *encoding* that represents the extent to which each basis vector is used to reconstruct each data vector.
- Alternatively, when rows in \mathbf{X} are data points in N -dimensional space, columns in \mathbf{V} correspond to basis vectors and each row in \mathbf{U} represents encoding.

Applying NMF to a term-document matrix for document clustering, each column of \mathbf{X} is treated as a data point in m -dimensional space. In such a case, the factorization (1) is interpreted as follows.

- U_{ij} corresponds to the degree to which term t_i belongs to cluster c_j . In other words column j of \mathbf{U} , denoted by \mathbf{u}_j , is associated with a prototype vector (center) for cluster c_j .
- V_{ij} corresponds to the degree document d_i is associated with cluster j . With appropriate normalization, V_{ij} is proportional to a posterior probability of cluster c_j given document d_i . More details on probabilistic interpretation of NMF for document clustering are summarized in Sec. 2.2.

2.1 Multiplicative Updates for NMF

We consider the squared Euclidean distance as a discrepancy measure between the data \mathbf{X} and the model \mathbf{UV}^\top , leading to the following least squares error function

$$\mathcal{E} = \frac{1}{2} \|\mathbf{X} - \mathbf{UV}^\top\|^2. \quad (2)$$

NMF involves the following optimization:

$$\arg \min_{\mathbf{U} \geq 0, \mathbf{V} \geq 0} \mathcal{E} = \frac{1}{2} \|\mathbf{X} - \mathbf{UV}^\top\|^2. \quad (3)$$

Gradient descent learning (which is additive update) can be applied to determine a solution to (3), however, nonnegativity for \mathbf{U} and \mathbf{V} is not preserved without further operations at iterations.

On the other hand, a multiplicative method developed in [6] provides a simple algorithm for (3). We give a slightly different approach from [6] to derive the same multiplicative algorithm. Suppose that the gradient of an error function has a decomposition that is of the form

$$\nabla \mathcal{E} = [\nabla \mathcal{E}]^+ - [\nabla \mathcal{E}]^-, \quad (4)$$

where $[\nabla \mathcal{E}]^+ > 0$ and $[\nabla \mathcal{E}]^- > 0$. Then multiplicative update for parameters Θ has the form

$$\Theta \leftarrow \Theta \odot \left(\frac{[\nabla \mathcal{E}]^-}{[\nabla \mathcal{E}]^+} \right)^{-\eta}, \quad (5)$$

where \odot represents Hadamard product (elementwise product) and $(\cdot)^{-\eta}$ denotes the elementwise power and η is a learning rate ($0 < \eta \leq 1$). It can be easily seen that the multiplicative update (5) preserves the nonnegativity of the parameter Θ , while $\nabla \mathcal{E} = 0$ when the convergence is achieved.

Derivatives of the error function (2) with respect to \mathbf{U} with \mathbf{V} fixed and with respect to \mathbf{V} with \mathbf{U} fixed, are given by

$$\nabla_{\mathbf{U}} \mathcal{E} = [\nabla_{\mathbf{U}} \mathcal{E}]^+ - [\nabla_{\mathbf{U}} \mathcal{E}]^- = \mathbf{UV}^\top \mathbf{V} - \mathbf{XV}, \quad (6)$$

$$\nabla_{\mathbf{V}} \mathcal{E} = [\nabla_{\mathbf{V}} \mathcal{E}]^+ - [\nabla_{\mathbf{V}} \mathcal{E}]^- = \mathbf{VU}^\top \mathbf{U} - \mathbf{X}^\top \mathbf{U}. \quad (7)$$

With these gradient calculations, the rule (5) with $\eta = 1$ yields the well-known Lee and Seung's multiplicative updates [6]

$$\mathbf{U} \leftarrow \mathbf{U} \odot \frac{\mathbf{XV}}{\mathbf{UV}^\top \mathbf{V}}, \quad (8)$$

$$\mathbf{V} \leftarrow \mathbf{V} \odot \frac{\mathbf{X}^\top \mathbf{U}}{\mathbf{VU}^\top \mathbf{U}}, \quad (9)$$

where the division is also an elementwise operation.

2.2 Probabilistic Interpretation and Normalization

Probabilistic interpretation of NMF, as in probabilistic latent semantic indexing (PLSI), was given in [7] where equivalence between PLSI and NMF (with I -divergence) was shown.

Let us consider the joint probability of term and document, $p(t_i, d_j)$, which is factorized by

$$\begin{aligned} p(t_i, d_j) &= \sum_k p(t_i, d_j | c_k) p(c_k) \\ &= \sum_k p(t_i | c_k) p(d_j | c_k) p(c_k), \end{aligned} \quad (10)$$

where $p(c_k)$ is the prior probability for cluster c_k . Elements of the term-document matrix, X_{ij} , can be treated as $p(t_i, d_j)$, provided X_{ij} are divided by $\mathbf{1}^\top \mathbf{X} \mathbf{1}$ such that $\sum_i \sum_j X_{ij} = 1$ where $\mathbf{1} = [1, \dots, 1]^\top$ with appropriate dimension.

Relating (10) to the factorization (11), U_{ik} corresponds to $p(t_i | c_k)$, representing the significance of term t_i in cluster c_k . Applying sum-to-one normalization to each column of \mathbf{U} , i.e., $\mathbf{U} \mathbf{D}_U^{-1}$ where $\mathbf{D}_U \equiv \text{diag}(\mathbf{1}^\top \mathbf{U})$, we have an exact relation

$$[\mathbf{U} \mathbf{D}_U^{-1}]_{ik} = p(t_i | c_k).$$

Assume that \mathbf{X} is normalized such that $\sum_i \sum_j X_{ij} = 1$. We define a scaling matrix $\mathbf{D}_V \equiv \text{diag}(\mathbf{1}^\top \mathbf{V})$. Then the factorization (11) can be rewritten as

$$\mathbf{X} = (\mathbf{U} \mathbf{D}_U^{-1}) (\mathbf{D}_U \mathbf{D}_V) (\mathbf{V} \mathbf{D}_V^{-1})^\top. \quad (11)$$

Comparing (11) with the factorization (10), one can see that each element of the diagonal matrix $\mathbf{D} \equiv \mathbf{D}_U \mathbf{D}_V$ corresponds to cluster prior $p(c_k)$. In the case of unnormalized \mathbf{X} , the prior matrix \mathbf{D} absorb the scaling factor. In practice, the data matrix does not have to be normalized in advance.

In a task of clustering, we need to calculate the posterior of cluster $p(c_k | d_j)$. Applying Bayes' rule, the posterior of cluster is given by the document likelihood and cluster prior probability. That is, $p(c_k | d_j)$ is given by

$$\begin{aligned} p(c_k | d_j) &\propto p(d_j | c_k) p(c_k) \\ &= [\mathbf{D} (\mathbf{V} \mathbf{D}_V^{-1})^\top]_{kj} \\ &= [(\mathbf{D}_U \mathbf{D}_V) (\mathbf{D}_V^{-1} \mathbf{V}^\top)]_{kj} \\ &= [\mathbf{D}_U \mathbf{V}^\top]_{kj}. \end{aligned} \quad (12)$$

It follows from (12) that $(\mathbf{V} \mathbf{D}_U)^\top$ yields the posterior probability of cluster, requiring the normalization of \mathbf{V} using the diagonal matrix \mathbf{D}_U . Thus, we assign document d_j to cluster k^* if

$$k^* = \arg \max_k [\mathbf{V} \mathbf{D}_U]_{jk}.$$

Document clustering by NMF was first developed in [4]. Here we use only different normalization and summarize the algorithm below.

Algorithm outline: Document clustering by NMF

1. Construct a term-document matrix \mathbf{X} .
2. Apply NMF to \mathbf{X} , yielding $\mathbf{X} = \mathbf{UV}^\top$.
3. Normalize \mathbf{U} and \mathbf{V} :

$$\begin{aligned} \mathbf{U} &\leftarrow \mathbf{U} \mathbf{D}_U^{-1}, \\ \mathbf{V} &\leftarrow \mathbf{V} \mathbf{D}_U, \end{aligned}$$

where $\mathbf{D}_U = \mathbf{1}^\top \mathbf{U}$.

4. Assign document d_j to cluster k^* if

$$k^* = \arg \max_k V_{jk}.$$

3 Orthogonal NMF for Document Clustering

Orthogonal NMF involves a decomposition (1) as in NMF but requires that \mathbf{U} or \mathbf{V} satisfies the orthogonality constraint such that $\mathbf{U}^\top \mathbf{U} = \mathbf{I}$ or $\mathbf{V}^\top \mathbf{V} = \mathbf{I}$ [8]. In this paper we consider the case where $\mathbf{V}^\top \mathbf{V} = \mathbf{I}$ is incorporated into the optimization (3). In such a case, it was shown that orthogonal NMF is equivalent to k -means clustering in the sense that they share the same objective function [9]. In this section, we present a new algorithm for orthogonal NMF with $\mathbf{V}^\top \mathbf{V} = \mathbf{I}$ where we incorporate the gradient on Stiefel manifold into multiplicative update.

Orthogonal NMF with $\mathbf{V}^\top \mathbf{V} = \mathbf{I}$ is formulated as following optimization problem:

$$\begin{aligned} \arg \min_{\mathbf{U}, \mathbf{V}} \mathcal{E} &= \frac{1}{2} \|\mathbf{X} - \mathbf{UV}^\top\|^2 \\ \text{subject to } \mathbf{V}^\top \mathbf{V} &= \mathbf{I}, \mathbf{U} \geq 0, \mathbf{V} \geq 0. \end{aligned} \quad (13)$$

In general, the constrained optimization problem (13) is solved by introducing a Lagrangian with a penalty term $\text{tr} \left\{ \Lambda (\mathbf{V}^\top \mathbf{V} - \mathbf{I}) \right\}$ where Λ is a symmetric matrix containing Lagrangian multipliers. Ding *et al.* [3] took this approach with some approximation, developing multiplicative updates.

Here we present a different approach, incorporating the gradient in a constraint surface on which $\mathbf{V}^\top \mathbf{V} = \mathbf{I}$ is satisfied, into (5). With \mathbf{U} fixed in (2), we treat (2) as a function of \mathbf{V} . Minimizing (2) where \mathbf{V} is constrained to the set of $n \times K$ matrices such that $\mathbf{V}^\top \mathbf{V} = \mathbf{I}$ was well studied in [10,11]. Here we incorporate nonnegativity constraints on \mathbf{V} to develop multiplicative updates

with preserving the orthogonality constraint $\mathbf{V}^\top \mathbf{V} = \mathbf{I}$. The constraint surface which is the set of $n \times K$ orthonormal matrices such that $\mathbf{V}^\top \mathbf{V} = \mathbf{I}$ is known as the Stiefel manifold [12].

An equation defining tangents to the Stiefel manifold at a point \mathbf{V} is obtained by differentiating $\mathbf{V}^\top \mathbf{V} = \mathbf{I}$, yielding

$$\mathbf{V}^\top \boldsymbol{\Delta} + \boldsymbol{\Delta}^\top \mathbf{V} = 0, \quad (14)$$

i.e., $\mathbf{V}^\top \boldsymbol{\Delta}$ is *skew-symmetric*. The canonical metric on the Stiefel manifold [11] is given by

$$g_c(\boldsymbol{\Delta}, \boldsymbol{\Delta}) = \text{tr} \left\{ \boldsymbol{\Delta}^\top \left(\mathbf{I} - \frac{1}{2} \mathbf{V} \mathbf{V}^\top \right) \boldsymbol{\Delta} \right\}, \quad (15)$$

whereas the Euclidean metric is given by

$$g_e(\boldsymbol{\Delta}, \boldsymbol{\Delta}) = \text{tr} \left\{ \boldsymbol{\Delta}^\top \boldsymbol{\Delta} \right\}. \quad (16)$$

We define the partial derivatives of \mathcal{E} with respect to the elements of \mathbf{V} as

$$[\nabla_V \mathcal{E}]_{ij} = \frac{\partial \mathcal{E}}{\partial V_{ij}}. \quad (17)$$

For the function \mathcal{E} (2) (with \mathbf{U} fixed) defined on the Stiefel manifold, the gradient of \mathcal{E} at \mathbf{V} is defined to be the tangent vector $\tilde{\nabla}_V \mathcal{E}$ such that

$$\begin{aligned} g_e(\nabla_V \mathcal{E}, \boldsymbol{\Delta}) &= \text{tr} \left\{ (\nabla_V \mathcal{E})^\top \boldsymbol{\Delta} \right\} \\ &= g_c(\tilde{\nabla}_V \mathcal{E}, \boldsymbol{\Delta}) \\ &= \text{tr} \left\{ (\tilde{\nabla}_V \mathcal{E})^\top \left(\mathbf{I} - \frac{1}{2} \mathbf{V} \mathbf{V}^\top \right) \boldsymbol{\Delta} \right\}, \end{aligned} \quad (18)$$

for all tangent vectors $\boldsymbol{\Delta}$ at \mathbf{V} .

Solving (18) for $\tilde{\nabla}_V \mathcal{E}$ such that $\mathbf{V}^\top \tilde{\nabla}_V \mathcal{E}$ is skew-symmetric yields

$$\tilde{\nabla}_V \mathcal{E} = \nabla_V \mathcal{E} - \mathbf{V}(\nabla_V \mathcal{E})^\top \mathbf{V}. \quad (19)$$

Thus, with partial derivatives in (7), the gradient in the Stiefel manifold is calculated as

$$\begin{aligned} \tilde{\nabla}_V \mathcal{E} &= (-\mathbf{X}^\top \mathbf{U} + \mathbf{V} \mathbf{U}^\top \mathbf{U}) - \mathbf{V}(-\mathbf{X}^\top \mathbf{U} + \mathbf{V} \mathbf{U}^\top \mathbf{U})^\top \mathbf{V} \\ &= \mathbf{V} \mathbf{U}^\top \mathbf{X} \mathbf{V} - \mathbf{X}^\top \mathbf{U} \\ &= [\tilde{\nabla}_V \mathcal{E}]^+ - [\tilde{\nabla}_V \mathcal{E}]^-. \end{aligned} \quad (20)$$

Invoking the relation (5) with replacing ∇_V by $\tilde{\nabla}_V$ yields

$$\mathbf{V} \leftarrow \mathbf{V} \odot \frac{\mathbf{X}^\top \mathbf{U}}{\mathbf{V} \mathbf{U}^\top \mathbf{X} \mathbf{V}}, \quad (21)$$

which is our ONMF algorithm. The updating rule for \mathbf{U} is the same as (8).

4 Experiments

We tested our orthogonal NMF algorithm on the six standard document datasets (CSTR, k1a, k1b, re0, and re1) and compared the performance with the standard NMF and the Ding *et al.*'s orthogonal NMF (DTPP)[\[3\]](#). We applied the stemming and stop-word removal for each dataset, and select 1,000 terms based on the mutual information with the class labels. Normalized-cut weighting [\[4\]](#) is applied to the input data matrix.

We use the accuracy to compare the clustering performance of different algorithms. To compute the accuracy, we first applied Kuhn-Munkres maximal matching algorithm [\[13\]](#) to find the appropriate matching between the clustering result and the target labels. If we denote the true label for the document n to be c_n , and the matched label \tilde{c}_n , the accuracy AC can be computed by

$$AC = \frac{\sum_{n=1}^N \delta(c_n, \tilde{c}_n)}{N},$$

where $\delta(x, y) = 1$ for $x = y$ and $\delta(x, y) = 0$ for $x \neq y$. Because the algorithms gave different results depending on the initial conditions, we calculated the mean of 100 runs for different initial conditions. Our orthogonal NMF algorithm gave better performance than the standard NMF and DTPP for the most of the datasets (Table [1](#)).

The orthogonality of the matrix V is also measured by using $\|V^T V - I\|$. The changes of the orthogonality over the iterations are measured and averaged for 100 trials. Our orthogonal NMF algorithm obtained better orthogonality than DTPP for the most of the datasets. The change of orthogonality for the CSTR dataset is shown in Fig. [1](#) for an example.

Table 1. Mean clustering accuracies (n=100) of standard NMF, Ding *et al.*'s orthogonal NMF (DTPP), and our orthogonal NMF (ONMF) for six document datasets

	NMF	DTPP	ONMF
ctr	0.7568	0.7844	0.7268
wap	0.4744	0.4281	0.4917
k1a	0.4773	0.4311	0.4907
k1b	0.7896	0.6087	0.8109
re0	0.3624	0.3384	0.3691
re1	0.4822	0.4452	0.5090

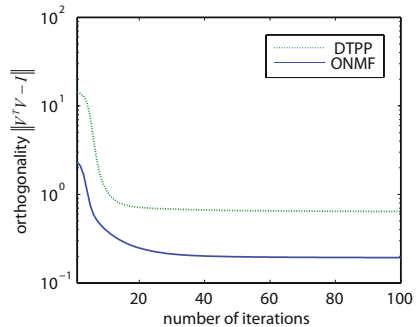


Fig. 1. The orthogonality $\|V^T V - I\|$ convergence of Ding *et al.*'s orthogonal NMF (DTPP) and our orthogonal NMF (ONMF) for the CSTR dataset

5 Conclusions

We have developed multiplicative updates on Stiefel manifold for orthogonal NMF and have successfully applied it to a task of document clustering, confirming its performance gains over standard NMF and existing orthogonal NMF.

Acknowledgments. This work was supported by National Core Research Center for Systems Bio-Dynamics and Korea Ministry of Knowledge Economy under the ITRC support program supervised by the IITA (IITA-2008-C1090-0801-0045).

References

1. Lee, D.D., Seung, H.S.: Learning the parts of objects by non-negative matrix factorization. *Nature* 401, 788–791 (1999)
2. Li, S.Z., Hou, X.W., Zhang, H.J., Cheng, Q.S.: Learning spatially localized parts-based representation. In: *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, Kauai, Hawaii, pp. 207–212 (2001)
3. Ding, C., Li, T., Peng, W., Park, H.: Orthogonal nonnegative matrix tri-factorizations for clustering. In: *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, Philadelphia, PA (2006)
4. Xu, W., Liu, X., Gong, Y.: Document clustering based on non-negative matrix factorization. In: *Proceedings of the ACM SIGIR International Conference on Research and Development in Information Retrieval (SIGIR)*, Toronto, Canada (2003)
5. Shahnaz, F., Berry, M., Pauca, P., Plemmons, R.: Document clustering using non-negative matrix factorization. *Information Processing and Management* 42, 373–386 (2006)
6. Lee, D.D., Seung, H.S.: Algorithms for non-negative matrix factorization. In: *Advances in Neural Information Processing Systems (NIPS)*, vol. 13. MIT Press, Cambridge (2001)
7. Gaussier, E., Goutte, C.: Relation between PLSA and NMF and implications. In: *Proceedings of the ACM SIGIR International Conference on Research and Development in Information Retrieval (SIGIR)*, Salvador, Brazil (2005)
8. Choi, S.: Algorithms for orthogonal nonnegative matrix factorization. In: *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, Hong Kong (2008)
9. Ding, C., He, X., Simon, H.D.: On the equivalence of nonnegative matrix factorization and spectral clustering. In: *Proceedings of the SIAM International Conference on Data Mining (SDM)*, Newport Beach, CA, pp. 606–610 (2005)
10. Smith, S.T.: *Geometric Optimization Methods for Adaptive Filtering*. Ph.D thesis, Harvard University (1993)
11. Edelman, A., Arias, T., Smith, S.T.: The geometry of algorithms with orthogonality constraints. *SIAM J. Matrix Anal. Appl.* 20(2), 303–353 (1998)
12. Stiefel, E.: Richtungsfelder und fernparallelismus in n-dimensionalem mannigfaltigkeiten. *Commentarii Math. Helvetici* 8, 305–353 (1935)
13. Lovasz, L., Plummer, M.: *Matching Theory*. Akademiai Kiado (1986)

Feature Discovery by Enhancement and Relaxation of Competitive Units

Ryotaro Kamimura

IT Education Center, Tokai University
1117 Kitakaname Hiratsuka Kanagawa 259-1292, Japan
ryo@cc.u-tokai.ac.jp

Abstract. In this paper, we introduce a new concept of *enhancement* and *relaxation* to discover features in input patterns in competitive learning. We have introduced mutual information to realize competitive processes. Because mutual information is an average over all input patterns and competitive units, it cannot be used to detect detailed feature extraction. To examine in more detail how a network is organized, we introduce the enhancement and relaxation of competitive units through some elements in a network. With this procedure, we can estimate how the elements are organized with more detail. We applied the method to a simple artificial data and the famous Iris problem to show how well the method can extract the main features in input patterns. Experimental results showed that the method could more explicitly extract the main features in input patterns than the conventional techniques of the SOM.

1 Introduction

The information-theoretic approach has been introduced in neural networks with many applications [1], [2], [3], [4], [5], [6]. In particular, mutual information has played important roles in the information-theoretic approach, because mutual information can be used to represent a degree of organization in a network. We have introduced mutual information as a measure of structure in competitive learning [7], [8]. However, because mutual information is an average information over all input patterns and competitive units, it has been difficult to extract detailed features in networks.

To examine in more detail how a network is organized, we examine information change in a network by enhancing or relaxing competitive units with some elements such as input units, competitive units and input patterns. By examining information change in competitive units with the elements, we can estimate how these elements are organized to produce final outputs. To change information content in competitive units, we use a concept of *enhancement* or *relaxation*. We can enhance competitive units by using some elements in a network. With this enhancement, competitive units respond explicitly to input patterns; that is, information about input patterns in competitive units is increased. On the other hand, with relaxation, competitive units tend to respond to input patterns uniformly; that is, no information about input patterns is stored in competitive

units. The difference between the original information and enhanced or relaxed information is called *enhanced information*. Enhanced information can be used to detect the roles of elements in a network and to interpret final representations obtained by learning.

2 Theory and Computational Methods

2.1 Enhancement and Relaxation

Sensitivity analysis [9, 10, 11, 12] has been well established in supervised learning, because one of the major problems of neural networks consists of the difficulty in interpreting final internal representations. However, there are few studies on unsupervised learning comparable to sensitive analysis in supervised learning, though competitive learning has been developed to discover main features in input patterns [13]. This is because it has been difficult to identify criteria comparable to those in the error terms between targets and outputs.

In this context, we introduce the enhancement and relaxation of competitive units for detailed feature detection. Figure 1 shows a process of enhancement and relaxation. Figure 1(a) shows an original situation obtained by competitive learning, in which three neurons are differently activated for input units. By using enhancement, in Figure 1(b), the characteristics of competitive unit activations are enhanced, and only one competitive unit is strongly activated. This means that obtained information in competitive units is larger. On the other hand, Figure 1(c) shows a state by relaxation, in which all competitive units respond uniformly to input units. Because competitive units cannot differentiate between input patterns, no information on input patterns is stored. Thus, enhancement

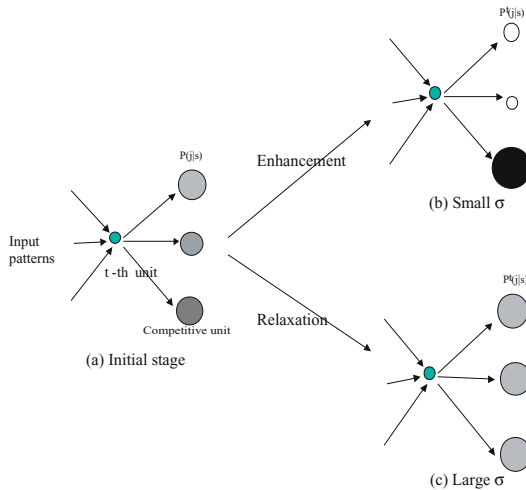


Fig. 1. Enhancement (b) and relaxation (c) in competitive learning

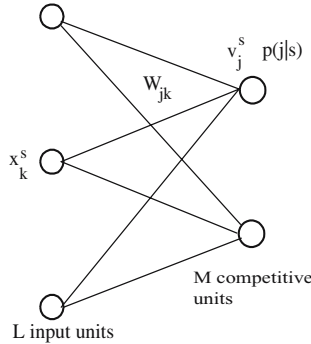


Fig. 2. A network architecture for competition

is used to increase information, and relaxation is used to decrease information about input patterns in competitive units.

2.2 Information Enhancement for Input Units

We examine whether information is changed by enhancing competitive units through some elements in competitive networks. We consider a network for competition, shown in Figure 2, in which x_k^s denotes the k th element of the s th input pattern, and w_{jk} represents a connection weight from the k th input unit to the j th competitive unit. Now, we focus upon the t th input unit and try to define enhanced information for the input unit. Distance when enhancement is realized by the t th input unit is defined by

$$d_{jt}^s = \sum_{k=1}^L \Phi_{kt}(x_k - w_{jk})^2, \tag{1}$$

where

$$\Phi_{kt} = \begin{cases} \frac{1}{\epsilon}, & \text{if } k = t; \\ \epsilon, & \text{otherwise.} \end{cases}$$

and where $0 < \epsilon < 1$ and L is the number of input units. By using this equation, we have competitive unit activations for the t th input unit

$$v_{jt}^s = \exp\left(-\frac{d_{jt}^s}{2\sigma^2}\right). \tag{2}$$

We can normalize these activations, and we have

$$p^t(j | s) = \frac{v_{jt}^s}{\sum_{m=1}^M v_{mt}^s}, \tag{3}$$

where M is the number of competitive units. The probability of the j th hidden unit is defined by

$$p^t(j) = \sum_{s=1}^S p(s)p^t(j | s), \tag{4}$$

where S is the number of input patterns. By using these probabilities, we have enhanced information when the t th input unit is deleted,

$$I_t = \sum_{s=1}^S \sum_{j=1}^M p(s)p(j | s) \log \frac{p(j | s)}{p^t(j | s)}. \quad (5)$$

2.3 Information Enhancement for Competitive Units

Then, we consider a case where enhancement is realized by a competitive unit. Competitive unit activations when the r th unit is used for enhancement is given by

$$d_{jr}^s = \Phi_{jr} \sum_{k=1}^L (x_k - w_{jk})^2, \quad (6)$$

where

$$\Phi_{jr} = \begin{cases} \frac{1}{\epsilon}, & \text{if } r = j; \\ \epsilon, & \text{otherwise.} \end{cases}$$

Finally, we have enhanced information for the r th competitive unit

$$I_r = \sum_{s=1}^S \sum_{j=1}^M p(s)p(j | s) \log \frac{p(j | s)}{p^r(j | s)}. \quad (7)$$

2.4 Information Enhancement for Input Patterns

Then, we consider a case where an input pattern is used for enhancement. Competitive unit activations when the q th input pattern is used for enhancement is given by

$$d_j^{sq} = \Phi^{sq} \sum_{k=1}^L (x_k - w_{jk})^2, \quad (8)$$

where

$$\Phi^{sq} = \begin{cases} \frac{1}{\epsilon}, & \text{if } q = s; \\ \epsilon, & \text{otherwise.} \end{cases}$$

By using these probabilities, we have enhanced information for the r th competitive unit

$$I_q = \sum_{s=1}^S \sum_{j=1}^M p(s)p(j | s) \log \frac{p(j | s)}{p^q(j | s)}. \quad (9)$$

3 Results and Discussion

In the following experiments, we try to show how well the new method extracts features in input patterns. Two experiments are simple enough to show the features extracted by the new method. For easy comparison, we use the conventional

SOM¹. All data in the following experiments were normalized with zero mean, and the variance was unity.

3.1 Artificial Data

In this experiment, we use the symmetric data shown in Figure 3. Because the data is symmetric in terms of input units, competitive units and input patterns, the same kinds of enhanced information for input units and input patterns are expected. Figure 3(b) shows a network architecture in which the number of input units is eight and the number of competitive units is 15 (3 by 5).

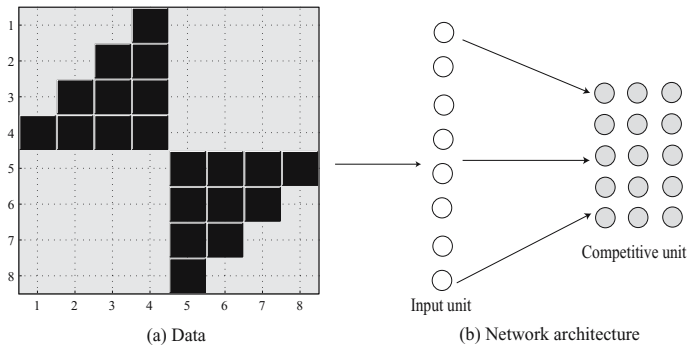


Fig. 3. Data (a) and a network architecture (b)

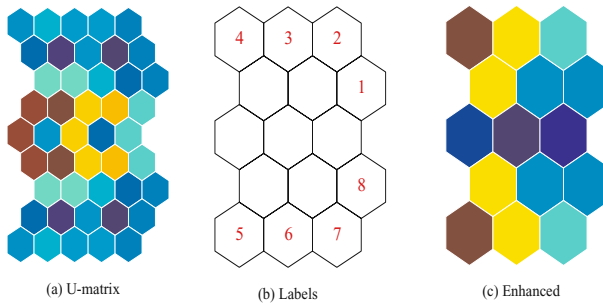


Fig. 4. U-matrix (a), a map with labels (b) and enhanced information (c) with $\sigma = 5$ and $\epsilon = 0.001$. Warmer and cooler colors show larger and smaller values.

Figure 4 shows a U-matrix (a), a map with labels (b) and enhanced information for competitive units (c). As shown in Figure 4(a), a clear boundary in brown can be seen in the middle of the U-matrix. Figure 4(b) shows that input

¹ We used SOM Toolbox 2.0, February 11th, 2000 by Juha Vesanto <http://www.cis.hut.fi/projects/somtoolbox/>. No special options were used for easy reproduction.

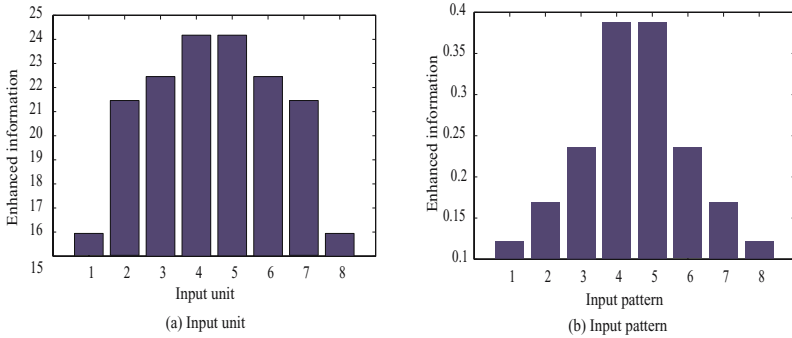


Fig. 5. Enhanced information for input units (a) and for input patterns (b) for an artificial data problem

patterns are located on a map with due consideration to distance among them. Figure 4(c) shows enhanced information for competitive units. We can see that a boundary in the middle is represented in dark blue. Neurons with large enhanced information are located on the corners of the left-hand side. As neurons are moved to the right, enhanced information becomes smaller. This observation corresponds perfectly to the labels on the map in Figure 4(c).

Figure 5(a) and (b) show enhanced information for input units and input patterns. As shown in the figures, enhanced information is larger when the input units and patterns are closer to the center. This corresponds to the characteristics of the data shown in Figure 4(a).

3.2 Iris Problem

In the second experiment, we use the famous Iris problem to show how well the new method captures the features in input patterns. The number of competitive units is 66 (6 by 11), which was given by the SOM software package. Figure 6 shows a U-matrix, a map with labels and enhanced information when the Gaussian width is 5 and the parameter ϵ is set to 0.001. As can be seen in Figure 6(a), the U-matrix shows a clear boundary in red or brown by which input patterns can be classified into two groups. However, the U-matrix does not show a boundary between classes 2 and 3, as shown in Figure 6(b). Figure 6(c) shows enhanced information. We can clearly see three groups in the map, and groups 2 and 3 are clearly separated. This result shows that enhanced information can extract features in input patterns more clearly.

In addition, we can see that the other types of enhanced information can be used to extract features corresponding to our intuition. Figure 7 shows enhanced information for input units (a) and input patterns (b). Figure 7(a) shows that a network tries to classify input patterns based upon features 3 and 4. Feature 3 especially, with the largest enhanced information, plays very important roles in classification. We can see in Figure 7(b) that enhanced information for input

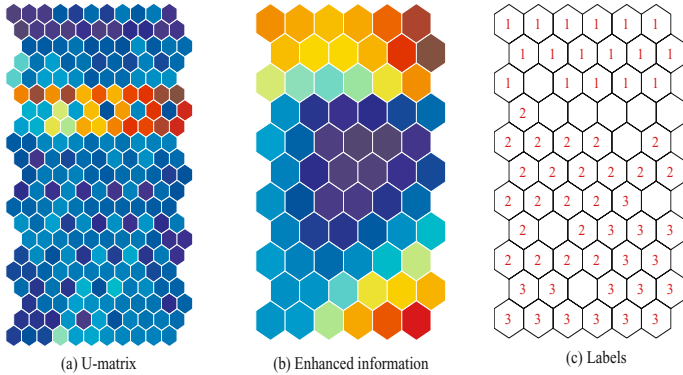


Fig. 6. U-matrix (a), a map with labels (b) and enhanced information (c) with $\sigma = 5$ and $\epsilon = 0.001$. Warmer and cooler colors represent larger and smaller values

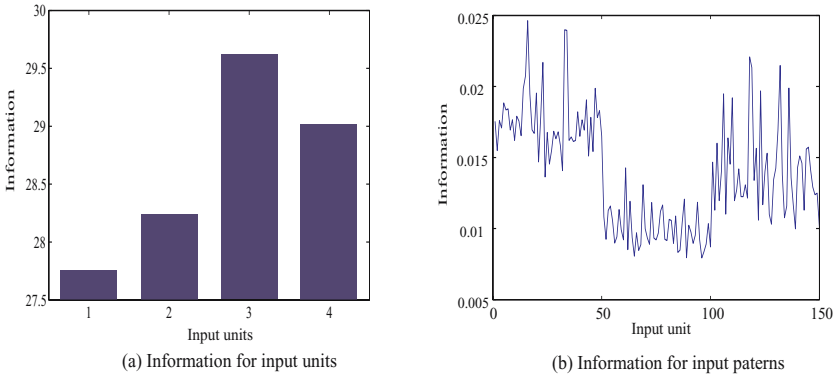


Fig. 7. Enhanced information for input units (a) and for input patterns (b) for the Iris problem

patterns clearly classifies the input pattern into three groups by its magnitude. These result show that enhanced information for input units and input patterns can be used to extract features in input patterns.

4 Conclusion

In this paper, we have introduced a new type of information called *enhanced information*. Enhanced information is obtained by enhancing competitive units through some elements in a network, while all the other competitive units are forced to be relaxed. If this enhancement causes a drastic change in information for competitive units, the elements surely play very important roles in information processing in competitive learning. We have applied the method to an

artificial data problem and the famous Iris problem to show how well the new method discovers features in input patterns. Experimental results have shown that features extracted by the new method are clearer than those by the conventional SOM, and results correspond to our intuition on input patterns.

We have used a pair of parameters in the paper. It is natural that final representations are greatly dependent upon combinations of the parameters. We should explore more exactly relations between final representations and the parameters. Though much study is needed for the new method to be practically applicable to many problems, we can say that our new method certainly shows a new direction for competitive learning.

References

1. Gokcay, E., Principe, J.: Information theoretic clustering. *IEEE Transactions on Pattern Analysis and Machine* 24(2), 158–171 (2002)
2. Lehn-Schioler, D.E.T., Hegde, A., Principe, J.C.: Vector-quantization using information theoretic concepts. *Natural Computation* 4(1), 39–51 (2004)
3. Torkkola, K.: Feature extraction by non-parametric mutual information maximization. *Journal of Machine Learning Research* 3, 1415–1438 (2003)
4. Linsker, R.: Self-organization in a perceptual network. *Computer* 21, 105–117 (1988)
5. Linsker, R.: How to generate ordered maps by maximizing the mutual information between input and output. *Neural Computation* 1, 402–411 (1989)
6. Bell, A.J., Sejnowski, T.J.: An information-maximization approach to blind separation and blind deconvolution. *Neural Computation* 7(6), 1129–1159 (1995)
7. Kamimura, R.: Information-theoretic competitive learning with inverse euclidean distance. *Neural Processing Letters* 18, 163–184 (2003)
8. Kamimura, R.: Unifying cost and information in information-theoretic competitive learning. *Neural Networks* 18, 711–718 (2006)
9. Mozer, M.C., Smolensky, P.: Using relevance to reduce network size automatically. *Connection Science* 1(1), 3–16 (1989)
10. Karnin, E.D.: A simple procedure for pruning back-propagation trained neural networks. *IEEE Transactions on Neural Networks* 1(2) (1990)
11. Le Cun, J.S.D.Y., Solla, S.A.: Optimal brain damage. In: *Advanced in Neural Information Processing*, pp. 598–605 (1990)
12. Reed, R.: Pruning algorithms-a survey. *IEEE Transactions on Neural Networks* 4(5) (1993)
13. Rumelhart, D.E., Zipser, D.: Feature discovery by competitive learning. *Cognitive Science* 9, 75–112

Genetic Feature Selection for Optimal Functional Link Artificial Neural Network in Classification

Satchidananda Dehuri¹, Bijan Bihari Mishra², and Sung-Bae Cho¹

¹ Soft Computing Laboratory, Department of Computer Science,
Yonsei University, 262 Seongsanno, Seodaemun-gu, Seoul 120-749, Korea
satchi.lapa@gmail.com, sbcho@yonsei.ac.kr

² Department of Computer Science and Engineering,
College of Engineering Bhubaneswar, Patia, 751024, Orissa, India
misrabijan@gmail.com

Abstract. This paper proposed a hybrid functional link artificial neural network (HFLANN) embedded with an optimization of input features for solving the problem of classification in data mining. The aim of the proposed approach is to choose an optimal subset of input features using genetic algorithm by eliminating features with little or no predictive information and increase the comprehensibility of resulting HFLANN. Using the functionally expanded selected features, HFLANN overcomes the non-linearity nature of problems, which is commonly encountered in single layer neural networks. An extensive simulation studies has been carried out to illustrate the effectiveness of this method over to its rival functional link artificial neural network (FLANN) and radial basis function (RBF) neural network.

Keywords: Classification, Data mining, Genetic algorithm, FLANN, RBF.

1 Introduction

For the past few years, there have been a lot of studies focused on the classification problem in the field of data mining [1,2]. The general goal of data mining is to extract knowledge from large gamut of data. The discovered knowledge should be predictive and comprehensible. Knowledge comprehensibility is usually important for at least two related reasons. First, the knowledge discovery process usually assumes that the discovered knowledge will be used for supporting a decision to be made by a human user. Second if the discovered knowledge is not comprehensible to the user, he/she will not be able to validate it, hindering the interactive aspect of the knowledge discovery process, which includes knowledge validation and refinement. In this work the proposed method for classification is given an equal importance to both predictive accuracy and comprehensibility. We are measuring comprehensibility of the proposed method by reducing the architectural complexity. As we know the architectural complexity of FLANN [3] is directly proportional to number of features and the functions considered for expansion of the given feature value. Therefore, for reducing the architectural complexity we first select a subset of features (i.e. feature selection [4]) and

then applying the usual procedure of function expansion and training by back propagation learning. As the selection and learning is accomplished by hybridization of FLANN with genetic algorithms (GAs) [5], we named this method as hybrid FLANN (HFLANN).

Neural networks [6] have emerged as an important tool for classification. Pao et al. [7] shows a direction that their proposed FLANN may be conveniently used for function approximation and can be extended for classification with faster convergence rate and lesser computational load than an multi-layer perceptron (MLP) structure. The FLANN is basically a flat network and the need of the hidden layer is removed and hence the learning algorithm used in this network becomes very simple. The functional expansion effectively increases the dimensionality of the input vector and hence the hyper planes generated by the FLANN provide greater discrimination capability in the input pattern space. Although many types of neural networks can be used for classification purposes [7], we choose radial basis function neural network and our previous work FLANN for classification [3] as the benchmark method for comparison.

2 Functional Link Artificial Neural Network

The FLANN architecture uses a single layer feed forward neural network by removing the concept of hidden layers. This may sound a little harsh at first, since it is due to them that non-linear input-output relationships can be captured. Encouragingly enough, the removing procedure can be executed without giving up non-linearity, provided that the input layer is endowed with additional higher order functionally expanded units of the given pattern. The weighted sum of the functionally expanded features is fed to the single neuron of the output layer. The weights are optimized by the gradient descent method during the process of training.

The set of functions considered for function expansion may not be always suitable for mapping the non-linearity of the complex task. In such cases few more functions may be incorporated to the set of functions considered for expansion of the input dataset. However, dimensionality of many problems itself are very high and further increasing the dimensionality to a very large extent may not be an appropriate choice. So, it prompts us a new research direction to design HFLANN. The HFLANN harness the power of genetic algorithms (GAs) to reduce the dimensionality of the problem.

3 Proposed Method

The proposed HFLANN is a single layer ANN with a genetically optimized set of features. It has the capability of generating complex decision regions by non-linear enhancement of hidden units referred to as functional links. Figure 1 shows the topological structure of the HFLANN. The proposed method is characterized by a set of FLANN with a different subset of features.

Let n be the number of original features of the data domain. The number of features selected to become a chromosome of the genetic population is d , $d \leq n$. The d varies from chromosomes to chromosomes of the genetic population (i.e. $1 \leq d \leq n$). For simplicity, let us see how a single chromosome with d features is working cooperatively for HFLANN.

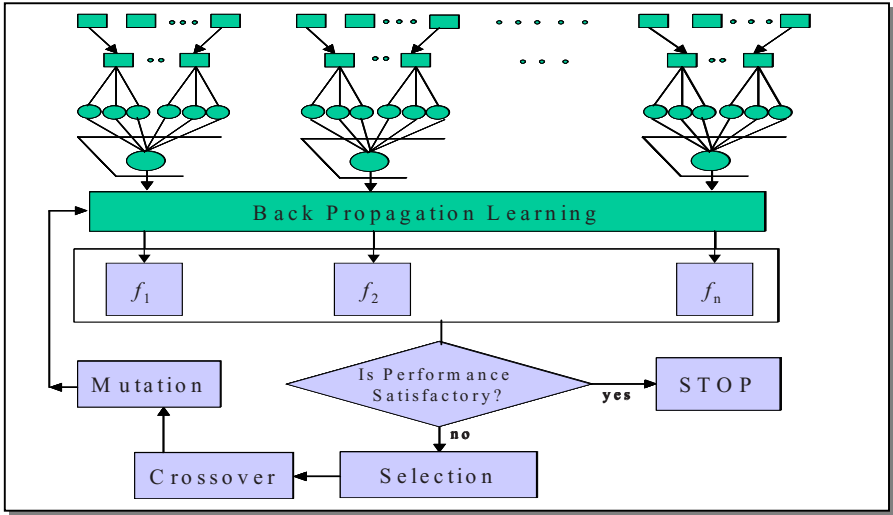


Fig. 1. Topological Structure of the HFLANN

In this work, we have used the general trigonometric function for mapping the d feature from low to high dimension. However, one can use a function that is very close to the underlying distribution of the data but it requires some prior domain knowledge. Here we are taking five functions out of which four are trigonometric and one is linear (i.e., keeping the original form of the feature value). Among the four trigonometric functions-two are sine and two are cosine functions. In the case of trigonometric functions the domain is feature values and range is a real number lies between $[-1,1]$. It can be written as

$$f : D \rightarrow R^{[-1,1] \cup \{x\}}, \tag{1}$$

where $D = \{x_{i1}, x_{i2}, \dots, x_{id}\}$, and d is known as the number of features.

In general let us take f_1, f_2, \dots, f_k be the number of functions used to expand each feature value of the pattern. Therefore, each input pattern can now be expressed as

$$\begin{aligned} \vec{x}_i &= \{x_{i1}, x_{i2}, \dots, x_{id}\} \\ &\rightarrow \{\{f_1(x_{i1}), f_2(x_{i1}), \dots, f_k(x_{i1})\}, \dots, \{f_1(x_{id}), f_2(x_{id}), \dots, f_k(x_{id})\}\}, \tag{2} \\ &= \{\{y_{11}, y_{21}, \dots, y_{k1}\}, \dots, \{y_{1d}, y_{2d}, \dots, y_{kd}\}\} \end{aligned}$$

The weight vector between hidden layer and output layer is multiplied with the resultant sets of non-linear outputs and are fed to the output neuron as an input. Hence the weighted sum is computed as follows:

$$s = \sum_{j=1}^m y_{ij} \cdot w_j, \quad i=1,2, \dots, N \text{ and } m \text{ be the total number of expanded features.} \tag{3}$$

The network has the ability to learn through back propagation learning. The training requires a set of training data, i.e., a series of input and associated output vectors. During the training, the network is repeatedly presented with the training data and the

weights adjusted by back propagation learning from time to time till the desired input-output mapping occurs. Hence, the estimated output is computed by the following metric:

$$\hat{y}_i(t) = f(s_i), i=1,2,\dots,N.$$

The error $e_i(t) = y_i(t) - \hat{y}_i(t)$, $i=1,2,\dots,N$ be the error obtained from the i^{th} pattern of the training set. Therefore, the error criterion function can be written as,

$$E(t) = \sum_{i=1}^N e_i(t), \quad (4)$$

and our objective is to minimize this function by gradient decent approach until $E \leq \varepsilon$.

This process is repeated for each chromosomes of the GA and subsequently each chromosome will be assigned a fitness value based on its performance. Using this fitness value the usual process of GA is executed until some good topology with an acceptable predictive accuracy is achieved.

3.1 High Level Algorithms for HFLANN

The specification of the near optimal HFLANN architecture and related parameters can be obtained by both genetic algorithms and back-propagation learning, as it is explained in the following. Evolutionary algorithms of genetic type are stochastic search and optimization methods. Principally based on computational models of fundamental process, such as reproduction, recombination and mutation. An algorithm of this type begins with a set (population) of estimates (genes), called individuals (chromosomes) appropriately encoded. Each one is evaluated for its fitness in solving the classification task of data mining. During each iteration (algorithm time-step) the most-fit individuals are allowed to make and bear offspring.

Individual Representation

For the evolutionary process the length of each particle is n (i.e. the upper bound of a feature vector). Each cell of the chromosome contains binary value either 0 or 1. The cell value controls the activation (the value of 1 is assigned) or deactivation (the value of 0 is assigned) of the functional expansion for individuals.

Objective Function

During evolution each individual measures its effectiveness by the error criterion function using equation (4) and the predictive accuracy is assigned as it corresponding fitness.

Pseudocode

The major steps of HFLANN can be described as follows:

1. DIVISION OF DATASET
Divide the dataset into two parts: training and testing
2. RANDOM INITIALIZATION
Initialize each individual randomly from the domain $\{0,1\}$.

3. REPEAT
4. FOR THE POPULATION
 - 4.1 FOR each sample of the training set
 - 4.2 MAPPING OF INPUT PATTERN

Map each pattern from low to high dimension, i.e. expand each feature value according to the predefined set of functions.
 - 4.3 CALCULATE the weighted sum and feed as an input to the node of the output layer.
 - 4.4 CALCULATE the error and accumulate it.
 - 4.5 BACK PROPAGATION LEARNING

Minimize the error by back propagation learning.
 - 4.6 ASSIGN THE FITNESS
5. FOR THE POPULATION
 - 5.1 Perform Roulette Wheel Selection to obtain the better chromosomes.
6. FOR THE POPULATION
 - 6.1 Perform recombination
 - 6.2 Mutation
7. UNTIL *< Maximum Iteration is Reached >*

4 Experimental Studies

The performance of the EFLANN model was evaluated using a set of ten public domain datasets like IRIS, WINE, PIMA, BUPA, ECOLI, GLASS, HOUSING, LED7, LYMPHOGRAPHY and ZOO from the University of California at Irvine (UCI) machine learning repository [8]. In addition we have taken VOWEL dataset to show the performance of HFLANN for classifying six overlapping vowel classes [9]. We have compared the results of HFLANN with other competing classification methods such as radial basis function network (RBF) and our previously proposed FLANN with gradient descent. Table 1 summarizes the main characteristics of the databases that have been used in this paper.

Table 1. Summary of the Dataset used in Simulation Studies

Sl. No.	Dataset	Instances	Attribute	Classes
1	IRIS	150	4	3
2	WINE	178	13	3
3	PIMA	768	8	2
4	BUPA	345	6	2
5	ECOLI	336	7	8
6	GLASS	214	9	6
7	VOWEL	871	3	6
8	HOUSING	506	13	5
9	LED7	UD	7	10
10	LYMPHOGRAPHY	148	18	4
11	ZOO	101	16	7

4.1 Parameter Setup

For evaluating the proposed algorithm, the following user defined parameters and protocols related to the dataset need to be set beforehand.

A two fold cross validation is carried out for all the dataset by randomly dividing the dataset into two parts (dataset1.dat and dataset2.dat). Each of these two sets was alternatively used either as a training set or test set.

The quality of each individual is measured by the predictive performance obtained during training. It is also very important to set the optimal values of the following parameters to reduce the local optimality. The parameters are described as follows:

Population size: The size of the population denoted as $|P|=50$ is fixed for all the datasets.

Length of the individuals is fixed to n , where n is the number of input features. The probability for crossover is 0.7 and mutation is 0.02. The number of iterations is 1000 for all the datasets.

4.2 Comparative Performance

The predictive performance obtained from HFLANN for the above datasets were compared with the results obtained from FLANN with back propagation learning and radial basis function network (RBF). Table 2 summarizes the average training and test performances of HFLANN and compared with FLANN and RBF.

Table 2. Average Comparative Performance of HFLANN, FLANN, and RBF

Dataset	Algorithms					
	HFLANN		FLANN		RBF	
	Training	Testing	Training	Testing	Training	Testing
IRIS	98.0001	97.3335	96.6665	96.6665	38.5000	38.5000
WINE	99.4380	90.4495	97.1910	88.7640	85.3935	79.2130
PIMA	80.7290	72.1355	79.5570	72.1355	77.4740	76.0415
BUPA	77.6820	69.2785	77.9725	69.2800	71.0125	66.9530
ECOLI	55.1670	50.8020	49.9625	47.3075	31.1780	26.1100
GLASS	63.5565	51.5075	60.7510	50.3800	48.9865	34.6440
VOWEL	40.4395	38.1965	27.9250	24.7220	25.2555	24.3250
HOUSING	82.2130	72.5295	76.4825	69.7630	67.1940	65.4150
LED7	30.8110	27.5280	22.4185	19.7000	20.2820	16.5720
LYMPHO.	97.2970	77.0270	91.8920	74.3245	85.1350	72.2927
ZOO	99.0385	86.1850	97.1155	85.1645	96.1540	81.0830

From Table 2, one can easily verify that except BUPA case in all other cases on an average the proposed method is giving promising results in both training and test cases. In the case of BUPA, FLANN is performing better. Figure 1 shows the percentage of feature selected by the HFLANN, which is very important in the context of comprehensibility. The X-axis represents the datasets and Y-axis represents the percentage of active bits in the optimal chromosome obtained during the training.

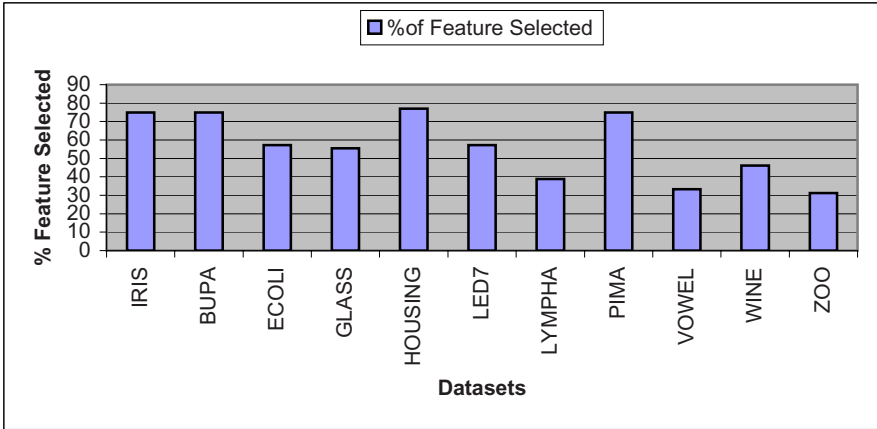


Fig. 2. Percentage of Optimal Set of Selected Features

4.3 Knowledge Incorporation in Predictive Accuracy

Let n be the total number of features in the dataset; T_1 and T_2 denote the number of feature selected using the training set 1 and testing set 2 alternatively.

Notations and their Meaning: $|N|$ represent the total number of features in the dataset, $|T_1|$ denote the total number of selected features in test set 2, $|T_2|$ denote the total number of selected features in test set 1.

The fitness of the chromosome with respect to T_1 is

$$f(T_1) = \frac{|PA| \times |N| - \tau \times |T_1|}{|N|} \tag{5}$$

Similarly the fitness of the chromosome with respect to T_2 is

$$f(T_2) = \frac{|PA| \times |N| - \tau \times |T_2|}{|N|} \tag{6}$$

where $|PA|$ represent the predictive accuracy and τ represent the tradeoff between two criteria and its value is 0.01.

Table 3. Predictive Accuracy of HFLAN by Knowledge Incorporation with $\tau = 0.01$

Dataset	N	P.A. Test Set 1 Chromosome	P.A. Test Set 2 Chromosome
IRIS	4	95.9925	97.3260
WINE	13	89.8826	91.0064
PIMA	8	71.6125	72.6548
BUPA	6	70.3343	68.2013
ECOLI	7	54.6939	46.8973
GLASS	9	57.1374	45.8642
VOWEL	3	34.5063	41.8733
HOUSING	13	67.9771	77.0673
LED7	7	19.6551	35.3923
LYMPH	18	78.3724	75.6721
ZOO	16	87.7494	84.6119

Table 3 shows the predictive accuracy using equation (5) and (6) of the HFLANN by incorporating a kind of knowledge of each chromosome optimally selected with respect to test set 1 and test set 2.

5 Conclusions

In this paper, we have evaluated the HFLANN for the task of classification in data mining by giving an equal importance to the selection of optimal set of features. The HFLANN model functionally maps the selected set of feature from lower to higher dimension. The experimental studies demonstrated that the classification performance of HFLANN model is promising. In almost all cases, the results obtained with the HFLANN proved to be better than the best results found by its competitor like RBF and FLANN with back propagation learning. The architectural complexity is low, whereas training time is little bit costly as compared to FLANN. As we know one of the most important criterions of data mining is how comprehensible the model is? If the architectural complexity increases than the comprehensibility decreases. Therefore, from this aspect we can claim that the proposed model can fit in data mining task of classification.

Acknowledgments. The authors would like to thank Department of Science and Technology, Govt. of INDIA and BK21 research program on Next Generation Mobile Software at Yonsei University, SOUTH KOREA for their financial support.

References

1. Kriegel, H.-P., et al.: Future Trends in Data Mining. *Data Mining and Knowledge Discovery* 15(1), 87–97 (2007)
2. Ghosh, A., Dehuri, S., Ghosh, S.: *Multi-objective Evolutionary Algorithms for Knowledge Discovery from Databases*. Springer, Heidelberg (2008)
3. Misra, B.B., Dehuri, S.: Functional Link Artificial Neural Network for Classification Task in Data Mining. *Journal of Computer Science* 3(12), 948–955 (2007)
4. Oh, I.-S., et al.: Hybrid Genetic Algorithms for Feature Selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20(11), 1424–1437 (2004)
5. Goldberg, D.E.: *Genetic Algorithms in Search, Optimization and Machine Learning*. Morgan Kaufmann, San Francisco (1989)
6. Zhang, G.P.: Neural Networks for Classification: A Survey. *IEEE Transactions on Systems, Man, Cybernetics-Part C: Application and Reviews* 30(4), 451–461 (2000)
7. Pao, Y.-H., et al.: Neural-net Computing and Intelligent Control Systems. *International Journal of Control* 56(2), 263–289 (1992)
8. Blake, C.L., Merz, C.J.: *UCI Repository of Machine Learning Databases*, <http://www.ics.uci.edu/~mllearn/MLRepository.html>
9. Pal, S.K., Majumdar, D.D.: Fuzzy Sets and Decision Making Approaches in Vowel and Speaker Recognition. *IEEE Transactions on Systems, Man, and Cybernetics* 7, 625–629 (1977)

A Novel Ensemble Approach for Improving Generalization Ability of Neural Networks

Lei Lu¹, Xiaoqin Zeng¹, Shengli Wu², and Shuiming Zhong¹

¹ Department of Computer Science and Engineering, Hohai University, Nanjing, China

² School of Computer and Mathematics, University of Ulster, UK

qingchun723@yahoo.com.cn, xzeng@hhu.edu.cn, s.wu1@ulster.ac.uk,
zhongyi@hhu.edu.cn

Abstract. Ensemble learning is one of the main directions in machine learning and data mining, which allows learners to achieve higher training accuracy and better generalization ability. In this paper, with an aim at improving generalization performance, a novel approach to construct an ensemble of neural networks is proposed. The main contributions of the approach are its diversity measure for selecting diverse individual neural networks and weighted fusion technique for assigning proper weights to the selected individuals. Experimental results demonstrate that the proposed approach is effective.

Keywords: Ensemble learning, diversity, sensitivity, fusion, clustering, the second training.

1 Introduction

In recent years, neural network ensemble [1] has been shown as an effective solution for difficult tasks by many researchers. Ensemble learning is to generate a set of learners, each of which is trained for the same task, and then combine them for gaining better performance. Hansen and Salamon [2] first proposed ensemble learning. Their work demonstrated that an ensemble could have generalization ability dramatically improved. In 1996, Sollich and Krogh [3] gave a definition for ensemble learning, which has been widely accepted. Nowadays, Ensemble learning has become a hot research topic in both neural networks and machine learning communities [4].

There are two main motivations for ensemble learning: one is to achieve high training accuracy and the other is to improve generalization ability. Valiant [5] proposed the framework of PAC (Probably Approximately Correct) learnability, which proved the equality of weak learning and strong learning. A strong learning algorithm is usually difficult to obtain, but ensembling a set of weak learners, trained by easily obtained weak learning algorithm, can be viewed as a way of upgrading weak learners to strong learners with higher training accuracy. As to generalization aspect, diversity is a key characteristic to get better generalization performance and combining a set of diverse learners can be thought of as a way of managing generalization limitations of individual learners. Conceptually, different learners, even though they are trained with the same training data, will give different outputs on untrained data, but combining

them may minimize the effect of possible errors because their cooperation may compensate each other if a proper fusion technique is adopted. Therefore, with individual learners' diversity on untrained data and a proper fusion technique, an ensemble of individual learners could have better generalization performance than that of an individual.

The idea of combining learners, which is trained yet accurate or generalized, has been achieved by many ensemble approaches. Bagging [6] and Boosting [7] are the two most prevailing approaches for constructing ensembles. Both approaches, in order to generate diverse individuals, work by taking a base learning algorithm and invoking it many times with different training sets. They have been shown to be very effective in improving neural network's accuracy and generalization ability.

It is well known that diversity and fusion are two key issues in ensemble learning. In this paper, aiming at improving the generalization performance of neural networks, we present a novel ensemble approach to deal with the two issues. For the selection of diverse individuals, a diversity measure is established by employing neural networks' output sensitivity [8]. As to the combination of the selected individuals, a weighted fusion technique is proposed by training the linearly combined individuals with the Least Mean Square (LMS) algorithm [9] to find proper weight for each individual. Experiments have been conducted and the results showed the effectiveness of the approach.

The rest of the paper is arranged as follows. The next section introduces the framework of our approach from a global point of view; the third section discusses in more detail some crucial techniques of our approach; the fourth section shows some experimental results, which demonstrate that both the diversity measure and the second training technique are effective for improving neural networks' generalization ability, and the final section gives the conclusion.

2 Ensemble Framework

By now, most of ensemble approaches are experimentally dependent. Commonly, a neural network ensemble is constructed in two steps, i.e., creation of ensemble members and fusion of the members. In our study, our ensemble approach can be divided into three steps. It first creates a pool of neural networks by training them with a given accuracy, then selects a subset of the most diverse neural networks from the pool, and finally combined the selected ones with a weighted fusion technique.

2.1 Creating a Pool of Individual Neural Networks

Individuals' diversity is critical for improving generalization ability in ensemble learning since it doesn't make any sense to combine identical individuals and the diversity may leave a chance for them to compensate one another after properly combined. Diverse neural networks can usually be obtained through the following ways: altering initial weights, altering architecture of neural networks and altering training data Sets.

2.2 Selecting Individual Neural Networks

Although the above ways can create diverse neural networks, they can't guarantee that. So, how to evaluate the diversity among the individuals in a pool of trained neural networks is crucial in ensemble learning. Most existing approaches, such as Bagging and Boosting, simply combine all of the trained ones. In order to avoid combining similar individuals in the pool, we have established a kind of diversity measure [10] to select the most diverse individuals from the pool. In our study, neural network's output sensitivity, which reflects the effects of neural network inputs' variation on its output, is employed as the diversity measure and defined as:

$$\Delta F(x) = F(x + \Delta x) - F(x) \quad (1)$$

where $F(x)$ and x represent the function established by a neural network and the input of the neural network separately. Eq. (1) is a general form of the sensitivity, which can be specialized with respect to certain neural network model and quantified with available computational technique. In this paper, the MLP (Multilayer perceptron) is taken as the target neural network and the partial derivatives of an MLP's output to its input at training samples are computed to constitute a quantified sensitivity value.

So, with a given pool of trained MLPs and the established diversity measure, members of the pool can be clustered by using K-means clustering algorithm into N groups based on each member's quantified sensitivity value. It is obvious that the MLPs in the same group have similar sensitivity value and thus less output diversity on the data near training samples. Therefore, it is reasonable to select one MLP from each of the N groups respectively, and take the selected ones to form an ensemble.

2.3 Combining Individual Neural Networks

There are two main approaches for combining neural networks: one is voting for classification, and the other is weighted averaging for regression. For regression, simple averaging has a drawback of being unable to reflect the differences of members' learning ability, because even if all members are well trained they are still likely to have different ability on the untrained data. In order to overcome this shortcoming, weighted averaging is introduced to reflect neural networks' different learning ability, i.e., different ability being assigned different weights. But how to assign a proper weight to each member in an ensemble is still an open question. In our study, a novel fusion technique is explored, which assigns individual weight by using the second training with LMS algorithm on a training data set that is different from those used for training the individuals.

3 Ensemble Techniques

This section introduces some concrete techniques for the implementation of our ensemble approach.

3.1 The Sensitivity of MLPs

Generally, an MLP’s sensitivity reflects the effects of its input variation on its output. So, it is conceptually rational to employ a kind of sensitivity as a tool for exploring output differences among different MLPs. By taking sigmoid function as MLPs’ activation function, the sensitivity we adopted is defined as the derivative of an MLP’s output to its input at a training sample [8], which can be expressed as:

$$S = F'(X) |_{x=x^*} = \left(\frac{\partial F}{\partial x_1}, \frac{\partial F}{\partial x_2}, \dots, \frac{\partial F}{\partial x_n} \right)^T |_{x=x^*} \tag{2}$$

where $F(X)$ represents the function of an MLP with input variable X , and x^* being a given input sample.

The computation of Eq (2) can be done by computing partial derivatives in a back propagation way, from the output layer, through hidden layers, and finally to input layer. Neurons on different layers have different computational formulae as follows.

$$\frac{\partial F(x)_i^L}{\partial x_k^{L-1}} = \frac{\partial F(x)_i^L}{\partial net_i^L} * \frac{\partial net_i^L}{\partial x_k^{L-1}} = (1 + F(x)_i^L)(1 - F(x)_i^L) * w_{ik}^L \tag{3}$$

where Eq (3) is for neuron i ($1 \leq i \leq n^L$) on output layer (layer L), in which k ($1 \leq k \leq n^{L-1}$) represents the number of neurons on layer $L-1$. Similarly, Eq (4) is for neurons on a hidden layer (layer l) with j ($1 \leq j \leq n^{l+1}$) represents the number of neurons on layer $l+1$. Eq (5) is for element i on input layer.

$$\begin{aligned} \frac{\partial F(x)_i^l}{\partial x_k^{l-1}} &= \sum_{j=1}^{n^{l+1}} \frac{\partial F(x)_j^{l+1}}{\partial x_i^{l+1}} * \frac{\partial F(x)_i^l}{\partial net_i^l} * \frac{\partial net_i^l}{\partial x_k^{l-1}} \\ &= \sum_{j=1}^{n^{l+1}} \frac{\partial F(x)_j^{l+1}}{\partial x_i^{l+1}} * (1 + F(x)_i^l) * (1 - F(x)_i^l) * w_{ik}^l \end{aligned} \tag{4}$$

$$\frac{\partial F(x)_i^0}{\partial x_i^0} = \sum_{j=1}^{n^1} \frac{\partial F(x)_j^1}{\partial x_i^0} \tag{5}$$

The above formulae can be used for computing an MLP’s sensitivity value only at each training sample. If a training set has n samples, then the sensitivity value of a particular sample i is $|S_i|$, and the sensitivity value for the MLP on the entire training set can be computed by $\sum_{i=1}^n |S_i|$.

3.2 The Architecture of an Ensemble

In our approach, an ensemble is a linear combination of its members with assigned weights, which can be regarded as a larger network composing all members with one more layer as output layer. The output layer has only one neuron with linear activation function, and so the output of the ensemble is $F(x) = \sum_{i=1}^n h_i * w_i$. Fig. 1 shows the architecture of an ensemble, in which h_i is the i th member, and w is the weight vector. Now, a question is how to find the proper weight for each member.

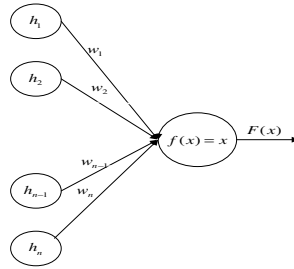


Fig. 1. The architecture of an ensemble

3.3 Fusion Weight Determination

One obvious solution for the above question is to find the weight by employing a training mechanism, called the second training. Under this consideration, the original training set needs to be divided into two parts, one (called *TRAINING SET I*) is for training each individual MLP, the other (called *TRAINING SET II*) is reserved for the second training to determine the fusion weights.

Since an ensemble can be treated as a one-layer network by regarding each member as an input element, the LMS algorithm can be employed for training the network on *TRAINING SET II*. But it should be noticed that LMS can not guarantee the network’s performance as each member does on *TRAINING SET I*. By noticing that all members are trained with approximate accuracy on *TRAINING SET I*, namely, $h_1 \approx h_2 \approx \dots \approx h_n$, the constraint of $\sum_{i=1}^n w_i = 1$ during the second training can overcome this problem because under this circumstance $F = \sum_{i=1}^n h_i * w_i \approx h_i * \sum_{i=1}^n w_i = h_i$ is always satisfied.

Thus, the LMS with the constraint $\sum_{i=1}^n w_i = 1$ can always make the ensemble have similar performance as individual members have on *TRAINING SET I*.

3.4 The Entire Procedure

The entire procedure of our approach can be summarized in Fig. 2.

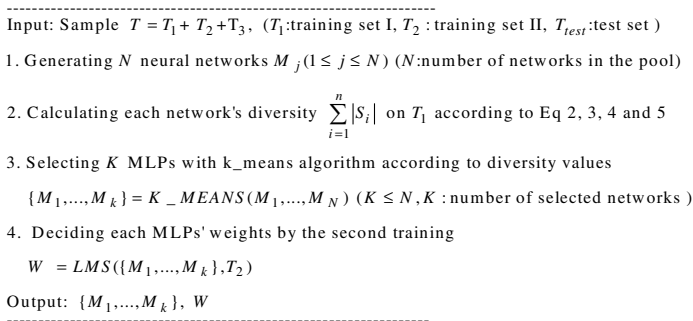


Fig. 2. The entire procedure of our approach

4 Experimental Results

In this section, to evaluate the effectiveness of our approach, two regression experiments are conducted on some well-known data sets and the experimental results are analyzed.

4.1 Experiment Setup

In our experiments, MLPs of two-layer architecture are implemented with five neurons on hidden layer and one neuron on output layer. Each data set in table 1 is divided into 3 subsets: *TRAINING SET I* for training individuals, *TRAINING SET II* for the second training and *TEST SET* for evaluating generalization performance. Each experiment runs 8 times and the MSE (Mean Squared Error) in each of the following tables is the average of the 8 results.

Table 1. Experiment Data

Data set	Function	Variable	Size
2d Mexican Hat	$y = \sin c x = \frac{\sin x }{ x }$	$x \in [-2\pi, 2\pi]$	5000[1]
			2500[2]
			2500[3]
3d Mexican Hat	$y = \sin c\sqrt{x_1^2 + x_2^2} = \frac{\sin\sqrt{x_1^2 + x_2^2}}{\sqrt{x_1^2 + x_2^2}}$	$x \in [-4\pi, 4\pi]$	3000[1]
			1500[2]
			1500[3]
Gabor	$y = \frac{1}{2}\pi \exp[-2(x_1^2 + x_2^2)] * \cos[2\pi(x_1^2 + x_2^2)]$	$x \in [0, 1]$	3000[1]
			1500[2]
			1500[3]
Plane	$y = 0.6x_1 + 0.3x_2$	$x \in [0, 1]$	1000[1]
			500[2]
			500[3]
Sin	$y = \sin x$	$x \in [-\pi, \pi]$	5000[1]
			2500[2]
			2500[3]
Sin1	$y = \left \frac{\sin x}{x} \right $	$x \in [-2\pi, 2\pi]$	5000[1]
			2500[2]
			2500[3]

In the above table, [1], [2] and [3] denote the size of *TRAINING SET I*, *TRAINING SET II* and *TEST SET* respectively.

4.2 Experiment I

In experiment I, 20 MLPs with different initial weights were trained on TRAINING SET I. They were then clustered into 9 groups by k-means algorithm according to each MLP’s sensitivity. One MLP was selected from each group respectively. All the selected MLPs are combined by the second training method.

The results of experiment I are shown in table 2, in which p_{20} is the generalization performance of an ensemble with 20 MLPs using simple averaging, $p_{\text{sin gle}}$ is the performance of the best individual MLP from the trained 20 MLPs, p_{selected} is the performance of an ensemble with the 9 selected MLPs using simple averaging and $p_{2\text{-train}}$ is the performance of an ensemble with the 9 selected MLPs using the weighted averaging.

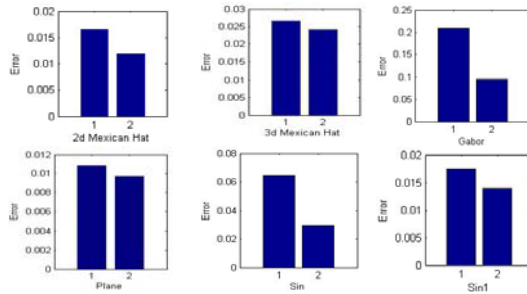
Table 2. The MSE on *TEST SET*

Data set	P_{20}	P_{single}	$P_{selected}$	$P_{2-train}$
2d Mex Hat	0.0155	0.0144	0.0154	0.0112
3d Mex Hat	0.0279	0.0275	0.0279	0.0265
Gabor	0.2148	0.2040	0.2145	0.1509
Plane	0.0116	0.0107	0.0115	0.0104
Sin	0.0668	0.0445	0.0646	0.0206
Sin1	0.0160	0.0148	0.0155	0.0129

As shown in the above tables, sensitivity measure is effective because $p_{selected}$ is better than p_{20} . In addition, the second training method for finding the appropriate fusion weights is also effective because $p_{2-train}$ has the best performance among P_{20} , P_{single} and $P_{selected}$.

4.3 Experiment II

In experiment II, 20 MLPs were generated by Bagging. Then, 9 of them were selected and combined by our approach. The performance, on *TEST SET*, of an ensemble with Bagging: $p_{bagging}$ as well as an ensemble with the second training method: $p_{bagging-2-train}$ is given in Fig. 3.

**Fig. 3.** The MSE of Bagging and our approach on *TEST SET*

In the above figure, 1 and 2 respectively denote the MSE of the ensemble with Bagging and that of our approach. It is clear that our approach has a better performance than Bagging.

5 Conclusion

This paper presents a novel approach for ensemble learning. To construct an ensemble of MLPs, a pool of trained MLPs is first created with different initial weights, then the

sensitivity of each MLP is computed to form a diversity measure for selecting diverse MLPs from the pool by means of a clustering method, finally each selected MLP's weight for combination is determined by the second training with LMS algorithm. Experimental results show that our approach can greatly improve an ensemble's generalization performance over the Bagging approach.

In our future work, we will try to proceed in three ways. They are the creation of more diverse individual neural networks satisfying the same training accuracy, the establishment of more accurate diversity measure as well as dynamic clustering method for selecting ensemble members, and the searching for more fitted weights for effectively combining individual members.

Acknowledgments. This work was supported by the National Natural Science Foundation of China under grants 60571048 and 60673186.

References

1. Dietterich, T.G.: Ensemble Learning The Handbook of Brain Theory and Neural Networks, 2nd edn. The MIT Press, Cambridge (2002)
2. Hansen, L.K., Salamon, P.: Neural Network ensembles. *IEEE Trans Pattern Analysis and Machine Intelligence* 12(10), 993–1001 (1990)
3. Sollich, P., Krogh, A.: Learning with ensembles: How over-fitting can be useful. In: Touretzky, D., Mozer, M., Hasselmo, M. (eds.) *Advances in Neural Information Processing Systems*, vol. 8, pp. 190–196. MIT Press, Cambridge (1996)
4. Sharkey, A.: *Combining Artificial Neural Nets: Ensemble and Modular Multi-Nets Systems*. Springer, London (1999)
5. Valiant, L.G.: A Theory of the learnable. *Communications of the ACM* 27(11), 1134–1142 (1984)
6. Breiman, L.: Bagging predictors. *Machine Learning* 24(2), 123–140 (1996)
7. Schapire, R.E.: The strength of weak learnability. *Machine Learning* 5(2), 197–227 (1990)
8. Tang, J., Zeng, X., Lu, L.: Ensemble Learning Based on the Output Sensitivity of Multilayer Perceptrons. In: *Proceedings of IEEE IJCNN*, pp. 1067–1072 (2007)
9. Hagan, M.T., Demuth, H.B., Beale, M.H.: *Neural Network Design*. PWS Publishing Company (1996)
10. Zeng, X., Yeung, D.S.: A Quantified Sensitivity Measure for Multilayer Perceptron to Input Perturbation. *Neural Computation* 15, 183–211 (2003)

Semi-supervised Learning with Ensemble Learning and Graph Sharpening

Inae Choi and Hyunjung Shin*

Department of Industrial & Information Systems Engineering, Ajou University
5 Wonchun-dong, Yeongtong-gu, Suwon, 443-749, Korea
{inae21, shin}@ajou.ac.kr

Abstract. The generalization ability of a machine learning algorithm varies on the specified values to the model-hyperparameters and the degree of noise in the learning dataset. If the dataset has a sufficient amount of labeled data points, the optimal value for the hyperparameter can be found via validation by using a subset of the given dataset. However, for semi-supervised learning--one of the most recent learning algorithms--this is not as available as in conventional supervised learning. In semi-supervised learning, it is assumed that the dataset is given with only a few labeled data points. Therefore, holding out some of labeled data points for validation is not easy. The lack of labeled data points, furthermore, makes it difficult to estimate the degree of noise in the dataset. To circumvent the addressed difficulties, we propose to employ ensemble learning and graph sharpening. The former replaces the hyperparameter selection procedure to an ensemble network of the committee members trained with various values of hyperparameter. The latter, on the other hand, improves the performance of algorithms by removing unhelpful information flow by noise. The experimental results present that the proposed method can improve performance on a publicly available bench-marking problems.

Keywords: Semi-supervised learning, Graph sharpening, Ensemble learning, Hyperparameter selection, Noise reduction.

1 Introduction

In supervised learning, the performance of a model would be improved if the data with class labels were more available, since the model would have more to learn. However, it is often difficult, expensive, and time-consuming to collect the data with labels while unlabeled data is readily available or relatively easy to collect such as in text categorization and protein function classification, etc. One may assume that those unlabeled data also give valuable information for learning. Recently, semi-supervised learning has been proposed to make use of unlabeled data as well as labeled ones by assuming that data with similar attributes lead to similar labels. Originally, this learning framework is to deal with situations where labeled data is only a few while unlabeled data is given in a large quantity. Previous researches have shown that learning with both unlabeled and labeled data can outperform learning with only labeled ones [1][2][3].

The generalization ability of a model, regardless of supervised learning or semi-supervised learning, varies on the specified values to model-hyperparameter and the degree of noise in the learning dataset. The hyperparameter selection depends on the degree of noise, and so the two problems have been often dealt with together as a single issue. However, either one can solely exist as a separate problem since, for instance, the hyperparameter selection will still remain even after the noisy data points are removed from the dataset. The hyperparameter selection is rather directly related to the complexity of problem in hand which is not likely to be identified in advance. If the dataset has a sufficient amount of labeled data, the optimal value for the hyperparameter can be found in a trial-and-error fashion by checking the validation performance. In supervised learning, cross-validation is generally used for this. However, for semi-supervised learning, it is hardly able to hold out some of labeled data in order to make a separate validation set. Meanwhile, noise in the dataset also increases the problem complexity. A higher degree of noise leads to a more complicated problem and hence a higher model complexity. If the degree of noise is known in advance, overfitting to noise can be prevented by imposing a more penalty on a more complicated model. In supervised learning, even if the degree of noise is not known a priori, the estimation for it is still available by checking the class impurity with labeled data. In semi-supervised learning, however, the noise estimation does not seem to be possible: again, because of lack of labeled data.

In this paper, we propose to employ ensemble learning and graph sharpening to circumvent the addressed difficulties. Ensemble learning is to combine a variety of models so as to improve performance by reducing the bias or variance of error [4][5][6][7]. And graph sharpening is a most recently proposed method in semi-supervised learning, which eliminates or reduces the noisy or corrupt information in the dataset by taking into explicit account the values of relationship between data points [8]. The former replaces the hyperparameter selection procedure to an ensemble network of the committee members trained with various values of hyperparameter. The latter, on the other hand, improves the performance of algorithms by removing or alleviating influence of noise in the dataset.

The paper is organized as follows. In Section 2, we present the basic idea of the proposed method with brief introduction to graph-based semi-supervised learning, graph sharpening, and ensemble learning. In Section 3, we show the results of experiments on synthetic and real-world datasets. We conclude with additional remarks in Section 4.

2 Method

The proposed method is based on graph-based semi-supervised learning. Within this framework, we employ graph sharpening and ensemble learning: First, multiple graphs are generated with various values of hyperparameter. The individual graphs are “sharpened” for de-noising. And then, those graphs are combined into an ensemble network. The following three subsections introduce the methods in due order.

2.1 Graph-Based Semi-supervised Learning

In graph-based semi-supervised learning [9], a data point $x_i (i=1, \dots, n)$ is represented as a node i in a graph, and the relationship between data points is represented by an edge (see Fig.1). The connection strength from each node j to each other node i is encoded in element w_{ij} of a weight matrix W . Often, a Gaussian function between points is used to specify connection strength:

$$w_{ij} = \begin{cases} \exp\left(-\frac{(x_i - x_j)^T (x_i - x_j)}{\sigma^2}\right) & \text{if } i \sim j, \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

The $i \sim j$ stands for node i and j having an edge between them that can be established by k nearest neighbors (kNN) where k is a user-specified hyperparameter. The labeled nodes have labels $y_i \in \{-1, 1\}$, while the unlabeled nodes have zeros $y_u = 0$. Our algorithm will output an n -dimensional real-valued vector $f = [f_1^T \dots f_n^T]^T = (f_1, \dots, f_1, f_{l+1}, \dots, f_{n=l+u})^T$, which can be thresholded to make label predictions on $f_{l=1}, \dots, f_n$ after learning. It is assumed that f_i should be closed to the given label y_i in labeled nodes (loss condition), and overall, f_i should not be too different from the f_j of adjacent nodes (smoothness condition). One can obtain f by minimizing the following quadratic function [9][10][11].

$$\min_f (f - y)^T (f - y) + \mu f^T L f \quad (2)$$

where $y = (y_1, \dots, y_l, 0, \dots, 0)^T$, and the matrix L , called the *graph Laplacian matrix*, is defined as $L=D-W$ where $D = \text{diag}(d_i)$, $d_i = \sum_j w_{ij}$. The parameter μ trades off loss versus smoothness. The solution of this problem is obtained as

$$f = (I + \mu L)^{-1} y \quad (3)$$

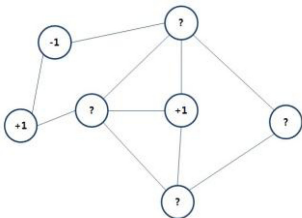


Fig. 1. An ordinary graph by W : The labeled node is denoted as “+1” or “-1”, and the unlabeled node as “?”. The edge has no directionality.

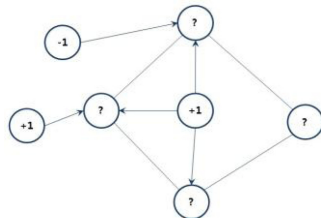


Fig. 2. A sharpened graph by W_s : By graph sharpening some edges have been removed or have assumed directionality according to the importance of the information flow

2.2 Graph Sharpening

The recently proposed graph sharpening is an effective method to improve the performance of the graph-based semi-supervised learning algorithms [8]. As described in the earlier section, the relationship between the data points is represented by the similarity matrix W which plays a critical role in prediction as a form of graph-Laplacian L . Related to the matrix, graph sharpening addresses two points. First, the data in many kinds of noise form an unnecessary edge and cause the decline of the performance of the algorithm. Second, the matrix W is dealt with as fixed and symmetric, which means the edge is considered without direction, and the reflected similarity is an undirected edge. When weight matrix W , however, describes the relationships between the labeled and unlabeled points, it is not necessarily desirable to regard all such relationships as symmetric. That is, the contribution of all edges to the information flow may be varied by not weighing them equally. Graph sharpening improves the performance of algorithms by changing the weight matrix to remove the edge caused by noise and to employ directionality between edges by asymmetrically weighing edge-weights [3]. If an ordinary weight matrix is represented as a block matrix $W = \begin{bmatrix} W_{ll} & W_{lu} \\ W_{ul} & W_{uu} \end{bmatrix}$, graph sharpening changes the matrix as $W = \begin{bmatrix} \text{diagonal} & 0 \\ W_{ul} & W_{uu} \end{bmatrix}$ in the simplest case. W_{lu} should be read as the weight of an edge from an unlabeled to a labeled point ($u \rightarrow l$). The output prediction for unlabeled data points can be obtained using the following equation:

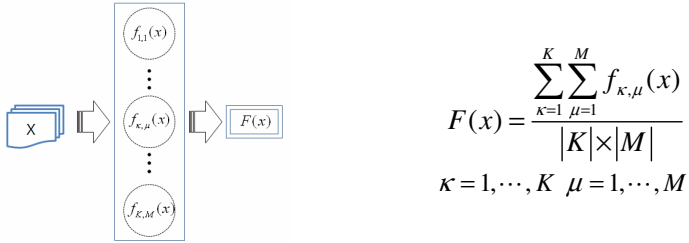
$$f_u = \mu(I + \mu(D_{uu} - W_{uu}))^{-1} W_{ul} y_l. \quad (4)$$

Fig. 2 shows change in a graph after sharpening. Note that some edges have been removed and have assumed directionality. A detailed mathematical foundation of graph sharpening can be found in [8].

2.3 Ensemble Learning

Ensemble learning is to combine a variety of member models in order to reduce the bias or variance of error, and to improve performance therefrom [4][5][6][7]. The ensemble network achieves the better performance when its members become more diverse. The members can be diversified by using perturbed learning datasets as in bagging [4] and boosting [12], or by directly perturbing the hyperparameter values of the learning algorithm while keeping a single learning dataset as common across the members [13]. In our method, the latter is taken- diversification by hyperparameter perturbation but no variation in the learning dataset. This becomes of great benefit to semi-supervised learning, particularly with respect to its hyperparameter selection procedure. Using a validation set for hyperparameter selection, the most representative approach which has originally been designed for supervised learning, does not well fit into semi-supervised learning. Because, in semi-supervised learning, the amount of labeled data in the entire dataset is absolutely deficient even for learning, in other words, even for making a training dataset, and so further splitting them for making a validation dataset is hard to be taken as a reasonable approach. In the proposed method, multiple networks are trained with various values of hyperparameter without

using a validation set, and then combined into an ensemble network: we train as many individual networks as all the possible combinations of the two hyperparameters, the number of neighbors $\kappa(\kappa=1, \dots, K)$ in Eq.(1) and the loss-smoothness tradeoff $\mu(\mu=1, \dots, M)$ in Eq.(2), and then the final output of the ensemble network is calculated by taking the simple mean of the output values of the $|K| \times |M|$ member networks. This replaces selecting a single best network via a validation set, which becomes a more practical approach for semi-supervised learning. Fig 3 illustrates the procedure.



$$F(x) = \frac{\sum_{\kappa=1}^K \sum_{\mu=1}^M f_{\kappa, \mu}(x)}{|K| \times |M|}$$

$\kappa = 1, \dots, K \quad \mu = 1, \dots, M$

Fig. 3. Ensemble learning

3 Experiment Results

We applied the proposed method, ensemble learning on sharpened graphs, to various kinds of data sets: an artificial dataset and five benchmarking datasets. We examined the change in the area under the ROC curve (AUC) in terms of ‘original’ versus ‘sharpened’ and ‘single’ versus ‘ensemble.’ This setting enabled us to see the effect of the proposed method from two separate viewpoints, graph sharpening and ensemble learning.

3.1 Artificial Data

The proposed method was evaluated on the two-moon toy problem as shown in Fig. 4(a). A total of 500 input data were generated from two classes, each with 245 unlabeled and 5 labeled data. The AUC was measured under various combinations of hyperparameters such as $(k, \mu) \in \{3, 5, 10, 20, 30\} \times \{0.01, 0.1, 1.0, 10, 100, 1000\}$, where k and μ indicate the number of k -nearest neighbors in Eq.(1) and the loss-smoothness tradeoff parameter in Eq.(2), respectively. Fig. 4(b) and (c) depict the changes in the AUC over the hyperparameter variation. Fig.4(b) shows the effect of graph sharpening: when compared with single-original, single-sharpened is less sensitive to hyperparameter variation because of noise reduction by graph sharpening. Also note that in every comparison the AUC is increased after the original graph is sharpened. Fig.4(c) shows that the synergy effect of graph sharpening and ensemble learning: when compared with single-original, ensemble-sharpened shows less sensitivity and higher accuracy. Also, when compared with single-sharpened, ensemble-sharpened gives a more stabilized performance.

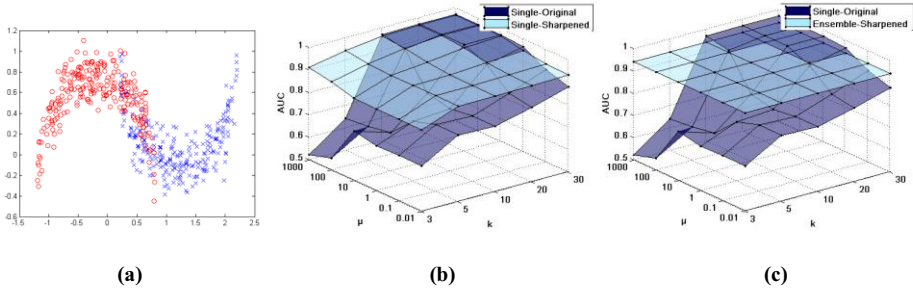


Fig. 4. Artificial data: (a) presents the two-moon toy problem. (b) and (c) depict the changes in the AUC over hyperparameter variation (k and μ), “(b) single-original vs. single-sharpened” and “(c) single-original vs. ensemble-sharpened”, respectively.

3.2 Real Data

Table 1 shows the AUC comparison results between “single-original” and “ensemble-sharpened” for five real-world datasets [15]. Each dataset has two sets of predetermined 12 splits, one is for 10 labeled data and the other is for 100 labeled data. The AUC was measured at every combination of hyperparameters (k, μ) $\in \{3, 5, 10, 20, 30\} \times \{0.01, 0.1, 1.0, 10, 100, 1000\}$. The Wilcoxon signed-rank test was used to verify the performances of both methods, where a smaller the value of p stands for a more significant difference between them [14]. The values listed in the table are the mean and standard deviation of AUC values across the 12 splits in datasets. Most in the cases, the proposed method increased AUCs and the effect was statistically significant. The maximum avg. increase in AUC, 0.10, was obtained from USPS-100 labeled dataset. On the other hand, the minimum avg. increase was 0 from BCI-100 labeled dataset guaranteeing ‘no loss’ even in the worst case. The five pairs of bar graphs in Fig.5 visualize the results.

Table 1. AUC comparison for the five benchmark data sets (mean \pm std)

Dataset (dimension, number of points)		Single-Original	Ensemble-Sharpned (proposed method)	p-value
(1)Digit1 (241, 1500)	10label	0.89 \pm 0.04	0.93 \pm 0.05	0.00
	100label	0.97 \pm 0.02	0.99 \pm 0.01	0.00
(2)USPS (241, 1500)	10label	0.65 \pm 0.09	0.68 \pm 0.10	0.00
	100label	0.87 \pm 0.08	0.97 \pm 0.01	0.00
(3)BCI (117,400)	10label	0.50 \pm 0.01	0.50 \pm 0.03	0.93
	100label	0.53 \pm 0.03	0.56 \pm 0.02	0.00
(4)g241c (241, 1500)	10label	0.55 \pm 0.03	0.56 \pm 0.05	0.00
	100label	0.63 \pm 0.05	0.65 \pm 0.04	0.00
(5)g241n (241, 1500)	10label	0.55 \pm 0.03	0.56 \pm 0.04	0.00
	100label	0.63 \pm 0.04	0.65 \pm 0.04	0.00

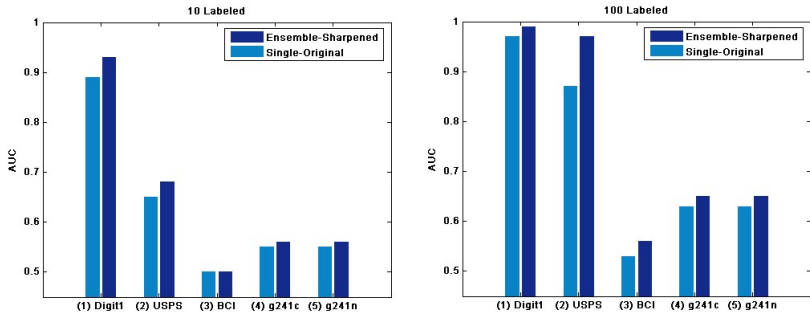


Fig. 5. The AUC comparison of single-original and ensemble-sharpened

4 Conclusion

In this paper, we proposed to use ensemble learning and graph sharpening for graph-based semi-supervised learning. Instead of using a single graph specified to a certain value of the hyperparameter in a trial-and-error fashion, we employed a graph ensemble of which committee members trained with various values of the hyperparameter. Ensemble learning stabilizes performance by reducing the error variance-and-bias of individual learners. Additionally, with ensemble learning, the hyperparameter selection procedure becomes less critical. The accuracy of an individual graph, on the other hand, was improved by the graph sharpening. Graph sharpening removes noisy or unnecessary edges from the original graph. This enhances the robustness to noise in the dataset. When applied to an artificial problem and five real-world problems, the synergy of ensemble learning and the graph sharpening resulted in more significant improvement in performance.

Acknowledgements

The authors would be like to gratefully acknowledge support from Post Brain Korea 21 and the research grant from Ajou University.

References

1. Zhu, X.: Semi-supervised learning with graphs. Ph.D. dissertation, Carnegie Mellon University (2005)
2. Shin, H., Tsuda, K.: Prediction of Protein Function from Networks. In: Chapelle, O., Schoelkopf, B., Zien, A. (eds.) Book: Semi-Supervised Learning, Ch. 20, pp. 339–352. MIT Press, Cambridge (2006)
3. Shin, H., Lisewski, A.M., Lichtarge, O.: Graph Sharpening plus Graph Integration: A Synergy that Improves Protein Functional Classification. *Bioinformatic* 23(23), 3217–3224 (2007)
4. Breiman, L.: Bagging Predictors. *Machine Learning* 24, 123–140 (1996)

5. Perrone, M.P.: Improving Regression Estimation: Averaging Methods for Variance Reduction with Extension to General Convex Measure Optimization. Ph.D Thesis, Brown University, Providence, RI (1993)
6. Sharkey, A.J.C.: Combining Diverse Neural Nets. *The Knowledge Engineering Review* 12(3), 231–247 (1997)
7. Tumer, K., Ghosh, J.: Error Correlation and Error Reduction in Ensemble Classifiers. *Connection Science* 8(3), 385–404 (1996)
8. Shin, H., Hill, N.J., Raetsch, G.: Graph-based semi-supervised learning with sharper edges. In: Fürnkranz, J., Scheffer, T., Spiliopoulou, M. (eds.) *ECML 2006*. LNCS (LNAI), vol. 4212, pp. 402–413. Springer, Heidelberg (2006)
9. Zhou, D., Bousquet, O., Lal, T.N., Weston, J., Schölkopf, B.: Learning with local and global consistency. *Advances in Neural Information Processing Systems (NIPS)* 16, 321–328 (2004)
10. Belkin, M., Matveeva, I., Niyogi, P.: Regularization and regression on large graphs. In: Shawe-Taylor, J., Singer, Y. (eds.) *COLT 2004*. LNCS (LNAI), vol. 3120, pp. 624–638. Springer, Heidelberg (2004)
11. Chapelle, O., Weston, J., Schölkopf, B.: Cluster kernels for semi-supervised learning. In: *Advances in Neural Information Processing Systems (NIPS)*, vol. 15, pp. 585–592 (2003)
12. Freund, Y., Schapire, R.E.: Experiments with a new boosting algorithm. In: *Machine Learning: Proceedings of the Thirteenth International Conference*, pp. 148–156 (1996)
13. Shin, H., Cho, S.: Pattern selection using the bias and variance of ensemble. *Journal of the Korean Institute of Industrial Engineers* 28(1), 112–127 (2001)
14. Demšar, J.: Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research* 7, 1–30 (2006)
15. <http://www.kyb.tuebingen.mpg.de/ssl-book/benchmarks.html>

Exploring Topology Preservation of SOMs with a Graph Based Visualization

Kadim Taşdemir^{1,*}

Yasar University, Computer Engineering
DYO Kampusu, Bornova, Izmir, Turkey
kadim.tasdemir@yasar.edu.tr

Abstract. The Self-Organizing Map (SOM), which projects a (high-dimensional) data manifold onto a lower-dimensional (usually 2-d) rigid lattice, is a commonly used manifold learning algorithm. However, a postprocessing – that is often done by interactive visualization schemes – is necessary to reveal the knowledge of the SOM. Thanks to the SOM property of producing (ideally) a topology preserving mapping, existing visualization schemes are often designed to show the similarities local to the lattice without considering the data topology. This can produce inadequate tools to investigate the detailed data structure and to what extent the topology is preserved during the SOM learning. A recent graph based SOM visualization, CONNvis [1], which exploits the underutilized knowledge of data topology, can be a suitable tool for such investigation. This paper discusses that CONNvis can represent the data topology on the SOM lattice despite the rigid grid structure, and hence can show the topology preservation of the SOM and the extent of topology violations.

1 Introduction

There have been many research on data projection and visualization motivated by the idea that the data samples can be represented by a low-dimensional submanifold even if they are high-dimensional. The visualization of the data space in low-dimensional (2-d or 3-d) spaces can guide the user for learning the underlying submanifold and the data structure. A classical technique is principal component analysis (PCA) which works well when the data lies on a linear submanifold of the data space. However, real data often lie on nonlinear spaces. For nonlinear projection, a number of algorithms have been introduced: multi dimensional scaling (MDS) [2], Isomap [3], locally linear embedding (LLE) [4], and the self-organizing maps (SOMs) [5] are some popular ones. A recent review on nonlinear projection schemes can be found in [6].

Among many schemes, SOMs stand out because they have two advantageous properties: an adaptive vector quantization that results in optimal placement of prototypes in the data space; and ordering of those prototypes on a rigid

* This work was partially supported by grant NNG05GA94G from the Applied Information Systems Research Program, NASA, Science Mission Directorate.

low-dimensional lattice according to their similarities. Due to these properties, density distribution – and therefore the structure – of a high-dimensional manifold can be mapped (and visualized) on a low-dimensional grid without reducing the dimensionality of the vectors.

SOMs, however, necessitate a postprocessing (visualization) scheme to reveal the learned knowledge due to the fact that the 2-d ordered placement of the quantization prototypes on the lattice is insufficient to show their similarities. What aspects of the SOM’s knowledge are presented by visualization has great importance for detailed analysis of the data structure. Various aspects of the SOM knowledge such as the (Euclidean) distances between the prototypes adjacent in the lattice and the density distribution are presented in several ways (different color assignments, adaptive cell sizes or cell shapes) [7,8,9,10,11,12]. More review on different SOM visualizations can be found in [13] and [14].

In order to visualize the data structure without postprocessing tools, Adaptive Coordinates [15] and the Double SOM [16] update not only the prototypes but also their positions in the SOM lattice while learning. By these methods, the SOM does not have a rigid grid anymore and the dissimilarities between the prototypes are visually exposed by the lattice distance of the prototypes. However, it is uncertain how they would work for large data volumes and for high-dimensional data. Another variant of the SOM that enables a direct and visually appealing measure of inter-point distances on the grid is the visualization induced SOM (ViSOM) [17]. The ViSOM produces a smooth and evenly partitioned mesh through the data points which reveals the discontinuities in the manifold. The ViSOM is computationally heavy for large data sets due to the requirement of large number of prototypes even for small data sets. To remedy this, a resolution enhanced version of ViSOM is also proposed [18].

Recently, a graph based topology visualization for SOMs, CONNvis, which renders the data topology – represented by a weighted Delaunay graph – on the SOM lattice is introduced [1]. It has been shown that CONNvis can help extraction of details in the data structure when the data vectors outnumber the SOM prototypes [14,19]. This paper demonstrates that when the SOM is used as a vector quantization algorithm CONNvis can be a useful tool for analysis of how data topology is mapped on the SOM lattice. Section 2 briefly introduces the CONNvis, shows examples for several different data sets, and compares the CONNvis to the U-matrix (a commonly used SOM visualization) [7] and to the ISOMAP (a popular MDS algorithm) [3]. Section 3 concludes the paper.

2 Data Topology Representation through CONNvis

CONNvis is a rendering data topology – represented by a “connectivity matrix” CONN – on the SOM lattice [14]. The connectivity matrix [14], CONN, is a weighted Delaunay triangulation, where the weight of an edge connecting two prototypes is the number of data vectors for which these prototypes are the best matching unit (BMU) and the second BMU. CONN indicates the neighborhood relations of the prototypes with respect to the data manifold because a binarized

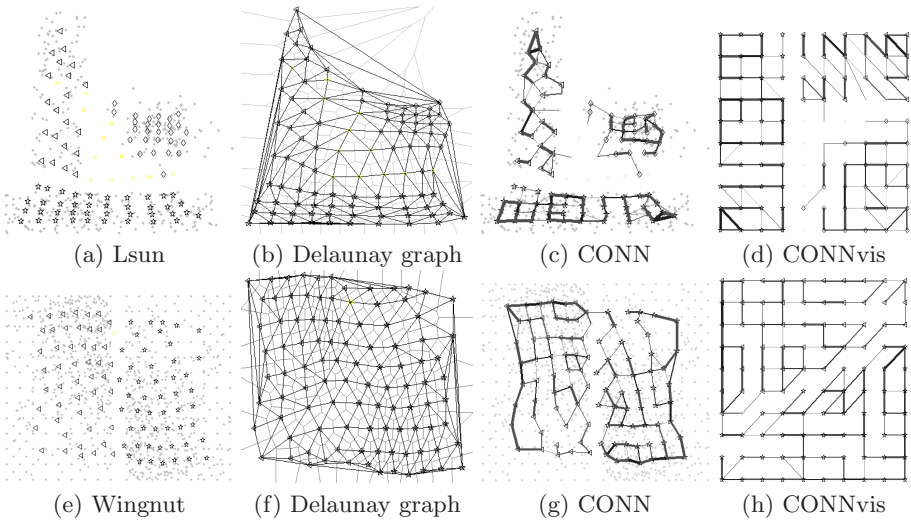


Fig. 1. Examples for the connectivity matrix CONN and its visualization, shown with two simple 2-d data sets from [21]. A 10×10 SOM is used to obtain prototypes. Top: (a) Lsun (dots, 3 clusters) and its prototypes in data space. (b) Delaunay triangulation of the Lsun prototypes. Empty prototypes are not shown. (c) CONN of Lsun prototypes. (d) CONNvis (Rendering of CONN on the SOM lattice). Bottom: (e) Wingnut (2 clusters with inhomogeneous density distribution) (f) Delaunay triangulation of the Wingnut prototypes (g) CONN of Wingnut prototypes. (h) CONNvis. The discontinuities in the Lsun and Wingnut data sets can be seen through their CONN and their CONNvis.

CONN is equivalent to the *induced* Delaunay graph, which is the intersection of the Delaunay triangulation with the data manifold [20]. This, in turn, makes the discontinuities (separations) within the data set visible. The weight of an edge, *connectivity strength*, show the degree of the similarity with respect to the data manifold.

Fig. 1 shows examples of CONN for two simple 2-d data sets constructed by [21]. The first one is called “Lsun” which has three well-separated clusters (two rectangular and one spherical). The second one, “Wingnut”, has two rectangular clusters with inhomogeneous density distribution within clusters and similar intra-cluster and inter-cluster distances. For both cases, the cluster structure (discontinuities in the data) can be seen through CONN regardless of the variations in cluster characteristics.

For low-dimensional (1-, 2-, 3-d) data sets, CONN can be visualized in the data space. However, for higher dimensions, it would not be practical to show CONN in the data space. In such cases, rendering of CONN on the SOM lattice can help visualize the data structure. This rendering, CONN visualization (CONNvis), is done by connecting the lattice locations of the prototypes with lines of various widths and colors. A line between two locations indicates that their prototypes are adjacent in the data space. The line width, which is proportional to connectivity

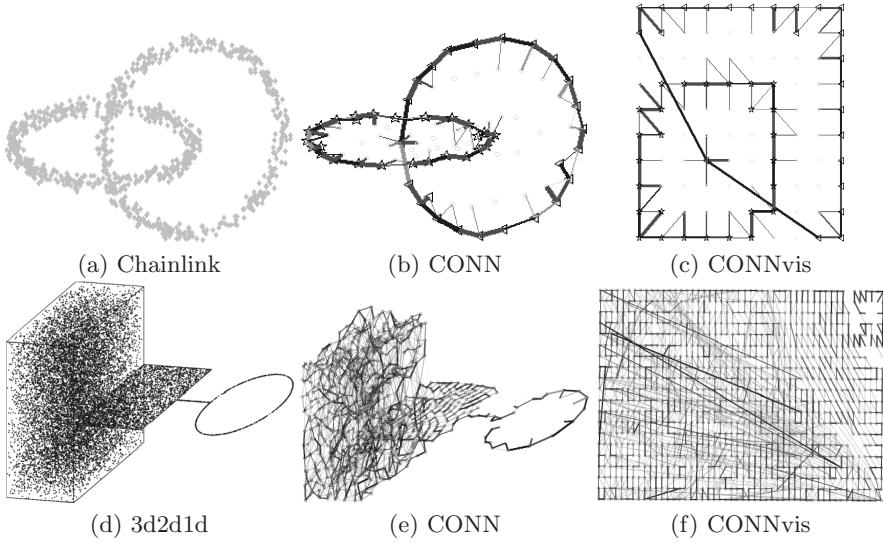


Fig. 2. Examples for topology representation with the connectivity matrix CONN and CONNvis, shown with two data sets. Top: Chainlink (a) data set (two circles in two different 2-d spaces) (b) CONN (c) CONNvis. Bottom: 3d2d1d (d) data (3-d rectangular, 2-d planar, 1-d line and 1-d circle) (e) CONN (f) CONNvis. For both data sets, CONN and CONNvis are able represent the data topology despite the different submanifolds in the data sets.

strengths, shows the local density distribution among the connected units. The line width hence conveys a sense of the *global importance* of each connection by allowing comparison with all others, in this visualization. The gray intensities (alternatively, different colors as in [11]) – dark to light – express the similarity ranking of the Voronoi neighbors of a prototype i in terms of the strengths of their connectivity to i . These intensities indicate the *local importance* of a connection since the ranking does not depend on the size of the receptive field of i , but only on the relative contribution of each neighbor. The existence of a line between two units, the line width and the gray intensity defines the similarity between the connected prototypes. As examples for CONNvis, CONN of the two data sets in Fig. 1 are rendered on the SOM lattice as shown in Fig. 1d and Fig. 1h. CONNvis representations of these data sets demonstrates the data topology just as informatively as CONN in the data space.

To illustrate the representation of data topology with the CONNvis, two more data sets which have different submanifolds are used. The first data set, Chainlink, is a 3-d data set with two rings lie in two different 2-d spaces. The second one, 3d2d1d, has a 3-d rectangular region, a 2-d planar region, a 1-d line and a circle. Fig. 2 shows these data sets, their CONN and CONNvis. Unlike other dimensionality reductions, SOM prototypes keep their data dimensionality while the prototypes are ordered on a 2-d grid. This allows visualizing the representation of the data topology in the data space and on the SOM. As seen in Fig. 2,

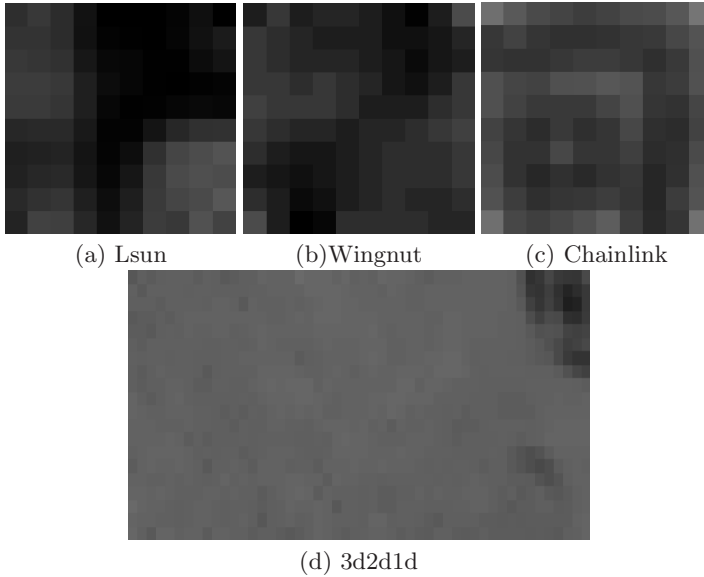


Fig. 3. U-matrix visualization of the SOMs for (a) Lsun in Fig. 1 (b) Wingnut in Fig. 1 (c) Chainlink in Fig. 2 (d) 3d2d1d in Fig. 2. The lighter the gray intensity, the more similar the neighbor prototypes are. Coarse boundaries of structures in the data can be seen through the U-matrix. However, finer details may be missed due to the lack of information about the data topology.

the data topology can be represented through the CONN and CONNvis. Even though this information is inherent in the SOM, other SOM visualizations may fail to represent the data topology for these cases due to their consideration of the prototype itself or its SOM neighbors. CONNvis uses the underutilized knowledge of the SOM and indicates that SOM can represent data topology to the extent that can show the separations within the data despite its rigid grid structure. The use of CONNvis aims to find the separations in the data rather than a perfect representation of the data topology.

2.1 Comparison of the CONNvis to the U-Matrix

A commonly used SOM visualization is the U-matrix [7]. The U-matrix shows the (average) distances of a prototype to its lattice neighbors by gray intensities of the cells. This visualization and its variants work well for relatively simple data. However, they may obscure finer details in complicated data and they underutilize the data topology. For example, Fig. 3 shows the U-matrix visualization of the SOM for the four data sets: Lsun, Wingnut, Chainlink, and 3d2d1d. The U-matrix visualizations of the 10×10 SOMs provide enough resolution for coarse delineation of the clusters of the Lsun and Wingnut data sets but may not show detailed boundaries of the clusters. A larger SOM (100×100) can discover these clusters through a U-matrix as shown in [10], at significantly

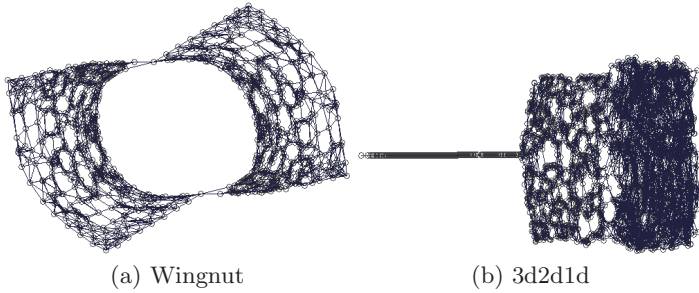


Fig. 4. ISOMAP mapping of (a) Wingnut in Fig. 1 (b) 3d2d1d in Fig. 2

larger computational cost. The two clusters of the Chainlink data set can be seen through the U-matrix, with the exception that one circle is captured as a line and the similarity of the two end points of this line could not be shown. For the case of the 3d2d1d data, the U-matrix indicates uniform distances among the prototypes that are lattice neighbors except for the top right of the SOM. This may indicate a different structure at the top right but details could not be identified from this visualization.

2.2 Comparison of the CONNvis to the ISOMAP

Another innovative method for data projection is ISOMAP [3]. It projects the data vectors onto a lower-dimensional space by preserving the relative local distances between data vectors. The ISOMAP projection of the Wingnut and 3d2d1d data sets onto 2-d space is shown in Fig. 4 (The Lsun and Chainlink data sets have disconnected groups and the ISOMAP embeds them separately and hence we omit them for ISOMAP representation). Similarly to the CONNvis, ISOMAP indicates the two clusters in the Wingnut data set and 3-d, 2-d and 1-d regions in the 3d2d1d data set, however, it misses the circle structure. ISOMAP often provides a better mapping than the SOM (hence a better 2-d visualization than CONNvis) when the data set has no discontinuities due to the fact that ISOMAP aims at finding one underlying submanifold. However, ISOMAP may be less useful than the CONNvis of the SOM in terms of visualization of discontinuities in the data.

2.3 Evaluation of Topology Violations with CONNvis

The representation of data topology on the SOM lattice with CONNvis also helps in a detailed assessment of topology preservation for the SOM learning. If two units are connected (their prototypes are Voronoi-neighbors) and they are neighbors in the SOM lattice, then the topology of their prototypes is preserved. However, there can be cases where connected units are not immediate SOM neighbors (*forward topology violations*) or unconnected units are immediate neighbors in the SOM lattice (*backward topology violations*). For example,

the CONNvis of the Lsun and Wingnut data sets, shown in Fig. 11, have no forward topology violations as expected since these data sets are 2-d. However, there are backward topology violations which reveal the discontinuities in the data such as the boundaries of three clusters in the Lsun and two clusters in the Wingnut data set. The CONNvis of the Chainlink data set (Fig. 12) shows two forward violations (the lines that connect the ends of the outer ring to the unit in the inner part) in the SOM whereas the CONNvis of the 3d2d1d data sets have several forward violations at the left part of the SOM where the 3-d manifold is mapped.

The strength (line width) of a forward topology violating connection characterizes the degree of the violation. The more data vectors contribute to a given connection, the more severe the violation is. For a forward violation, low strength (thin lines) usually indicates outliers or noise while greater strengths are due to data complexity or badly formed SOM. For example, the violations in the CONNvis of Chainlink and 3d2d1d are due to data complexity. The *folding length* of the violating connection, that is the maximum norm distance between the connected neural units in the SOM lattice, describes whether the topology violation is local (short ranged) or global (long ranged).

In most cases, perfect topology preservation is not necessary for capture of clusters in the data. Weak global violations, or violations that remain within clusters do not affect the delineation of boundaries. Proper investigation of such conditions for a trained SOM is therefore important. The CONNvis is a useful tool for such analysis. Interactive clustering from the CONNvis is explained and shown powerful in [14].

3 Conclusion

CONNvis is a postprocessing tool for the SOM where the number of data vectors is much larger than the SOM prototypes (due to the construction of CONN based on the density distribution). CONNvis drapes the data topology over the SOM lattice, which in turn, may help analyze the mapping of the SOM, topology preservation of the SOM, and the extent of forward topology violation at each prototype. By comparing the CONNvis to the U-matrix and the ISOMAP, it has been shown here that CONNvis can visualize the power of the SOMs in topology preservation despite the rigid SOM lattice.

References

1. Taşdemir, K., Merényi, E.: Data topology visualization for the Self-Organizing Maps. In: Proc. 14th European Symposium on Artificial Neural Networks (ESANN 2006), Bruges, Belgium, D-Facto, April 26-28, pp. 277–282 (2006)
2. Cox, T.F., Cox, M.: Multidimensional Scaling. Chapman and Hall/CRC, Boca Raton (2001)
3. Tenenbaum, J.B., de Silva, V., Langford, J.: A global geometric framework for nonlinear dimensionality reduction. *Science* 290(5500), 2319–2323 (2000)

4. Roweis, S., Saul, L.: Nonlinear dimensionality reduction by locally linear embedding. *Science* 290(5500), 2323–2326 (2000)
5. Kohonen, T.: *Self-Organizing Maps*, 2nd edn. Springer, Heidelberg (1997)
6. Yin, H.: Nonlinear multidimensional data projection and visualisation. In: Liu, J., Cheung, Y.-m., Yin, H. (eds.) *IDEAL 2003*. LNCS, vol. 2690, pp. 377–388. Springer, Heidelberg (2003)
7. Ultsch, A.: Self-organizing neural networks for visualization and classification. In: Lausen, O.B., Klar, R. (eds.) *Information and Classification-Concepts, Methods and Applications*, pp. 307–313. Springer, Berlin (1993)
8. Kraaijveld, M., Mao, J., Jain, A.: A nonlinear projection method based on Kohonen's topology preserving maps. *IEEE Trans. on Neural Networks* 6(3), 548–559 (1995)
9. Cottrell, M., de Bodt, E.: A Kohonen map representation to avoid misleading interpretations. In: *Proc. 4th European Symposium on Artificial Neural Networks (ESANN 1996)*, Bruges, Belgium, D-Facto, pp. 103–110 (1996)
10. Ultsch, A.: Maps for the visualization of high-dimensional data spaces. In: *Proc. 4th Workshop on Self-Organizing Maps (WSOM 2003)*, vol. 3, pp. 225–230 (2003)
11. Kaski, S., Kohonen, T., Venna, J.: Tips for SOM processing and colourcoding of maps. In: Deboeck, T.K.G. (ed.) *Visual Explorations in Finance Using Self-Organizing Maps*, London (1998)
12. Himberg, J.: A SOM based cluster visualization and its application for false colouring. In: *Proc. IEEE-INNS-ENNS International Joint Conf. on Neural Networks*, Como, Italy, vol. 3, pp. 587–592 (2000)
13. Vesanto, J.: SOM-based data visualization methods. *Intelligent Data Analysis* 3(2), 111–126 (1999)
14. Taşdemir, K., Merényi, E.: Exploiting data topology in visualization and clustering of Self-Organizing Maps. *IEEE Transactions on Neural Networks* (submitted)
15. Merkl, D., Rauber, A.: Alternative ways for cluster visualization in Self-Organizing Maps. In: *Proc. 1st Workshop on Self-Organizing Maps (WSOM 2005)*, Espoo, Finland, Helsinki University of Technology, June 4-6, pp. 106–111. Helsinki University of Technology, Neural Networks Research Centre, Espoo, Finland (1997)
16. Su, M.-C., Chang, H.-T.: A new model of self-organizing neural networks and its applications. *IEEE Transactions on Neural Networks* 12(1), 153–158 (2001)
17. Yin, H.: ViSOM- a novel method for multivariate data projection and structure visualization. *IEEE Transactions on Neural Networks* 13(1), 237–243 (2002)
18. Yin, H.: Resolution enhancement for the ViSOM. In: *Proc. 4th Workshop on Self-Organizing Maps (WSOM 2003)*, pp. 208–212 (2003)
19. Merényi, E., Taşdemir, K., Farrand, W.: Intelligent information extraction to aid science decision making in autonomous space exploration. In: *Proc. SPIE Defense and Security*, Orlando, Florida, March 17-20 (2008)
20. Martinetz, T., Schulten, K.: Topology representing networks. *Neural Networks* 7(3), 507–522 (1993)
21. Ultsch, A.: Clustering with som: U*c. In: *Proc. 5th Workshop on Self-Organizing Maps (WSOM 2005)*, Paris, France, September 5-8, 2005, pp. 75–82 (2005)

A Class of Novel Kernel Functions

Xinfei Liao¹ and Limin Tao²

¹ Department of Computer, Wenzhou Vocational and Technical College,
325035 Wenzhou Zhejiang, China

² School of Information Science and Engineering, Hangzhou Normal University,
310036 Hangzhou Zhejiang, China
{XinfeiLiao, LiminTao, liaoxinfei}@yahoo.cn

Abstract. This paper proposes a kind of novel kernel functions obtained from the reproducing kernels of Hilbert spaces associated with special inner product. SVM with the proposed kernel functions only need less support vectors to construct two-class hyperplane than the SVM with Gaussian kernel functions, so the proposed kernel functions have the better generalization. Finally, SVM with reproducing and Gaussian kernels are respectively applied to two benchmark examples: the well-known Wisconsin breast cancer data and artificial dataset.

Keywords: SVM, Reproducing Kernel.

1 Introduction

For the sake of the reproducing kernel functions of reproducing kernel Hilbert space with many good properties, they have been widely applied to many fields [1]-[3]. Nonlinear SVM based on kernel technique is a state-of-the-art learning machine which have been extensively used as classification and regression tool, and found a great deal of success in many applications [4]-[9]. In this paper, we prove that the reproducing kernel functions of reproducing kernel Hilbert space associated with novel inner product are a kind of new kernel functions. We apply SVM with the proposed kernel functions to Wisconsin breast cancer data and artificial data, and demonstrate that it provides remarkable improvement of support vectors and training time compared with that of SVM with the Gaussian kernels. Especially, the proposed kernel functions become more and more efficient with the increase of orders of the space.

2 Preliminaries

Definition 1. Let X be an abstract set, and H be a Hilbert space of real or complex value functions f defined on the set X . A function $K(x, y)$ on $X \times X$ is called a reproducing kernel, if $K(x, y)$ satisfies the following two properties:

- a) $K(x, \cdot)$ belongs to H for all $x \in X$;
- b) $\langle f(y), K(x, y) \rangle_y = f(x)$ for all $x \in X$ and all $f \in H$.

We denote δ as the Dirac function, F as the Fourier transformation and F^{-1} as the Fourier inverse transformation in this paper.

We denote $H^n(R)$ as the set of real functions that are absolutely continuous and square-integrable, with absolutely continuous derivatives up to the order $n-1$ and square-integrable derivatives up to the order n on R , where n is a positive integer.

We consider the Hilbert space $H^n(R)$ associated with the inner product

$$\langle u, v \rangle_{H^n(R)} = \int_R \sum_{i=0}^n C_i u^{(i)} v^{(i)} dx, \tag{1}$$

where $u(x)$ and $v(x) \in H^n(R)$, $C_i (i=0,1,2,\dots,n)$ is the coefficient of expansion of $(a+b)^n$.

$H^1(R)$ s are the reproducing kernel spaces [1].

$K_1(x) = \frac{1}{2} e^{-|x|}$ is the reproducing kernel of $H^1(R)$ associated with the inner product $\langle u, v \rangle = \int_R (uv + u'v') dx$ [2].

$$K_2(x) = (K_1 * K_1)(x) = \frac{1}{4} \int_R e^{-|y|-|x-y|} dy = \frac{1}{4} (1+|x|) e^{-|x|}. \tag{2}$$

$K_2(x)$ is the reproducing kernel of $H^2(R)$ associated with the inner product

$\langle u, v \rangle = \int_R (uv + 2u'v' + u''v'') dx$ [2].

$$K_3(x) = (K_1 * K_2)(x) = \frac{1}{8} \int_R (1+|y|) e^{-|y|-|x-y|} dy = \frac{1}{16} (x^2 + 3|x| + 3) e^{-|x|}. \tag{3}$$

$K_3(x)$ is the reproducing kernel of $H^3(R)$ associated with the inner product

$\langle u, v \rangle = \int_R (uv + 3u'v' + 3u''v'' + u'''v''') dx$ [2].

$$K_n(x) = (K_1 * K_{n-1})(x) = \dots = \underbrace{(K_1 * K_1 * \dots * K_1)}_{n \text{ times}}(x). \tag{4}$$

$K_n(x)$ is the reproducing kernel of $H^n(R)$ [1].

Theorem 1. The horizontal floating function is a allowable support vector’s kernel function if and only if the Fourier transform of $K(x)$ need satisfy the condition as follows:

$$F[K(x)] = (2\pi)^{-\frac{d}{2}} \int_{R^d} \exp(-j\omega x) K(x) dx \geq 0. \tag{5}$$

More details can be referred to [4].

3 Main Results

Theorem 2. If the reproducing kernel function is redefined as:

$$K(X, Y) = \prod_{i=1}^d K_n\left(\frac{x_i - y_i}{a_i}\right), \tag{6}$$

then it is a allowable support vector kernel function, where

$$X = (x_1, \dots, x_d)^T, \quad Y = (y_1, \dots, y_d)^T, \quad a_i > 0, \quad d \in \mathbb{Z}^+ . \tag{7}$$

Proof. According to the theorem 1, we only need to prove

$$F[K(X)] = (2\pi)^{-\frac{d}{2}} \int_{\mathbb{R}^d} \exp(-j\omega X) K(X) dX \geq 0 . \tag{8}$$

Substituting (6) into (8), the latter becomes

$$F[K(X)] = (2\pi)^{-\frac{d}{2}} \prod_{i=1}^d a_i \int_{-\infty}^{+\infty} K_n\left(\frac{x_i}{a_i}\right) \exp(-j\omega a_i \frac{x_i}{a_i}) d \frac{x_i}{a_i} = (2\pi)^{-\frac{d}{2}} \prod_{i=1}^d \frac{a_i}{(1 + \omega^2 a_i^2)^n} \tag{9}$$

Accordingly, we have

$$F[K(X)] > 0 . \tag{10}$$

4 Experiments

To evaluate the performance of $K_n(x)$, we did simulations on two classification problems: artificial dataset and Wisconsin breast cancer data classification problem. The primary kernel function is fixed to be a Gaussian RBF ($\sigma = 1$).

4.1 Artificial Datasets

Let us consider an artificial two-dimensional data set $X = \{(x, y)\}$ uniformly distributed in the region $[-1,1] \times [-1,1]$, where two classes are separated by a nonlinear boundary determined by $y = \sin \pi x$. Simulation results for $K_n(x)$ and Gaussian kernel are compared in the table 1.

Table 1. Comparison about the results for $K_n(x)$ and Gaussian kernel

Kernel function	Training errors	Test errors	SV's (the number of support vectors)
$K_1(x)$	0	7	11
$K_2(x)$	0	4	10
$K_3(x)$	0	3	8
Gaussian kernel	0	7	14

We should note that, since the data set is randomly generated, results in the test errors and SV's may change in different trials. However, the above results indicate the success of the method.

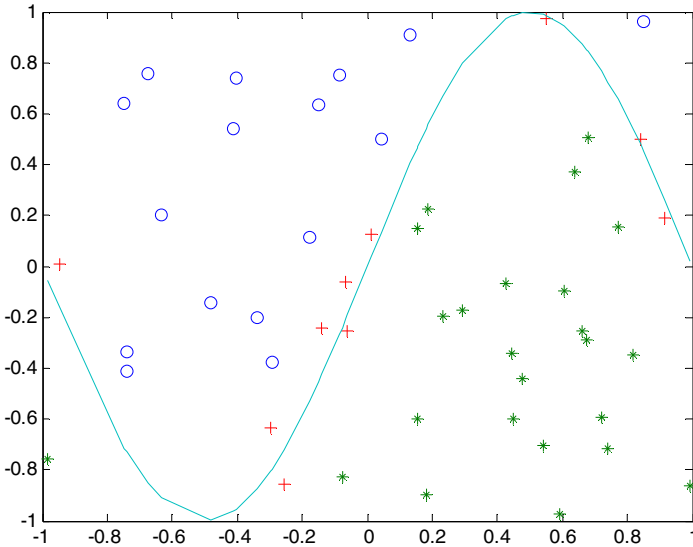


Fig. 1. A two-dimensional artificial data set, where the two classes are denoted by 'o' and '*'. The symbol '+' represents the support vectors of kernel K .

4.2 Wisconsin Breast Cancer Data Classification Problem

In this section, a benchmark Wisconsin breast cancer data classification problem is tested [10]. The data consists of 10 medical attributes (one of them is the id number which we don't use for the classification task), which are used to make a binary decision on the medical condition: whether the cancer is malignant or benign. The data set consists of 699 instances including missing values. We used a random selection of 200 training data and 200 testing data, excluding the instances with missing values. Experimental results for $K_n(x)$ and Gaussian kernel are compared in the table 2 which again demonstrates the power of $K_n(x)$.

Table 2. Comparison about the results for $K_n(x)$ and Gaussian kernel

Kernel function	Training samples	Test errors	SV's
$K_1(x)$	200	9	37
$K_2(x)$	200	9	34
$K_3(x)$	200	9	29
Gaussian kernel	200	9	42

Obviously, compared with Gaussian kernels, in terms of training samples and test errors under the same conditions, $K_n(x)$ need fewer support vectors, which requires less time. With the increase of n , the number of support vector is also declining.

5 Conclusions and Future Works

We prove that $K_n(x)$'s are a class of new kernel functions. We also achieve tremendous success in applying $K_n(x)$'s as the kernel functions of SVM. Experimental results show $K_n(x)$'s as the kernel functions of SVM are better than Gaussian kernel functions. With the increase of n , $K_n(x)$'s became better and better. Because $K_n(x)$ is remarkably analogous with the radial basis functions, wavelets and other kernel functions, we are exploring to substitute $K_n(x)$ for the radial basis functions, wavelets and other kernel functions. We believe that $K_n(x)$ is very attractive so that it can be effectively used to solve many hard kernel-based problems.

References

1. Zhang, Q.L., Wang, S.T.: A solution to the reproducing kernels space. In: 5th International DCABES Conference, pp. 42–46. Shanghai University Press, Shanghai (2006)
2. Qinli, Z., et al.: The numerical methods for solving Euler system of equations in reproducing kernel space $H^2(\mathbb{R})$. *Journal of Computational Mathematics* 3, 327–336 (2001)
3. Xianggen, X., Nashed, M.Z.: The Backus-Gilbert methods for signals in reproducing kernel Hilbert spaces and wavelet subspaces. *Inverse Problems* 10, 785–804 (1994)
4. Sánchez, V.D.A.: Advanced support vector machines and kernel methods. *Neurocomputing* 2, 5–20 (2003)
5. Evgeniou, T., Pontil, M., Poggio, T.: Regularization networks and support vector machines. *Advance Computational Math.* 1, 1–50 (2000)
6. Smola, A., Scholkopf, B., Muller, K.R.: The connection between regularization operators and support vector kernels. *Neural Networks* 11, 637–649 (1998)
7. Gunter, L., Zhu, J.: Efficient Computation and Model Selection for the Support Vector Regression. *Neural Computation* 19, 1633–1655 (2007)
8. Burges, C.J.C.: A tutorial on SVM for pattern recognition. *Data mining and Knowledge Discovery* 2, 955–974 (1998)
9. Chapelle, O., Vapnik, V., Bengio, Y.: Model Selection for Small Sample Regression. *Machine Learning* 48, 9–23 (2002)
10. Zhang, Q.L., Wang, S.T.: A Novel SVM and its application Breast Cancer. In: 1st IEEE ICBBE, pp. 633–636. IEEE Press, New York (2007)

RP-Tree: A Tree Structure to Discover Regular Patterns in Transactional Database*

Syed Khairuzzaman Tanbeer, Chowdhury Farhan Ahmed, Byeong-Soo Jeong,
and Young-Koo Lee

Department of Computer Engineering, Kyung Hee University
1 Sochun-ri, Kihung-eup, Youngin-si, Kyonggi-do, South Korea, 446-701
{tanbeer, farhan, jeong, yklee}@khu.ac.kr

Abstract. Temporal regularity of pattern appearance can be regarded as an important criterion for measuring the interestingness in several applications like market basket analysis, web administration, gene data analysis, network monitoring, and stock market. Even though there have been some efforts to discover *periodic* patterns in time-series and sequential data, none of the existing works is appropriate for discovering the patterns that occur regularly in a transactional database. Therefore, in this paper, we introduce a novel concept of mining *regular* patterns from transactional databases and propose an efficient data structure, called Regular Pattern tree (RP-tree in short), that enables a pattern growth-based mining technique to generate the complete set of *regular* patterns in a database for a user-given *regularity* threshold. Our comprehensive experimental study shows that RP-tree is both time and memory efficient in finding *regular* pattern.

Keywords: Data mining, pattern mining, regular pattern, cyclic pattern.

1 Introduction

Mining interesting patterns from transactional database plays an important role in data mining and knowledge engineering research. Different forms of interestingness have been investigated and several techniques have been developed in each case. Mining frequent patterns [1, 2] that discovers a set of frequently appearing patterns in a database has been studied widely in recent years. However, the number of occurrences may not always represent the significance of a pattern. The other important criterion for identifying the interestingness of a pattern might be the shape of occurrence. Consider the transactional database in Table 1. It can be observed that patterns “*a*”, “*d*” and “*be*” with respective supports 5, 5 and 4 only appear more frequently at a certain part of database (i.e., “*a*” at the beginning, “*d*” at the end, and “*be*” in the middle of database) than the rest part. Even though such patterns may be frequent in the whole database, their appearance behaviors do not follow temporal regularity. In contrast, relatively less frequent patterns “*c*”, “*bc*”, “*ce*”, “*ef*” etc. are almost evenly distributed

* This study was supported by a grant of the Korea Health 21 R&D Project, Ministry for Health, Welfare and Family Affairs, Republic of Korea (A020602).

Table 1. A transactional database

Id	Transaction	Id	Transaction	Id	Transaction
	<i>a d</i>	4	<i>a b c e</i>	7	<i>c d e</i>
	<i>a b c e</i>	5	<i>a b e f</i>	8	<i>d e f</i>
	<i>a b e f</i>	6	<i>b c d</i>	9	<i>b c d</i>

throughout the database. Therefore, they can be more important patterns in terms of the regularity of appearance which traditional frequent pattern mining techniques fail to discover. We define such a pattern that appears regularly in a database as a *regular* pattern.

The significance of such patterns with temporal regularity can be revealed in a wide range of applications where users might be interested on the occurrence behavior (*regularity*) of patterns rather than just occurring frequency i.e., support. For example, in a retail market some products may be sold more regularly than other products. That is, it is necessary to find out a set of items that are sold together at a regular interval. Also, to improve web site design the site administrator may be interested in regularly visited web page sequences rather than heavily hit web pages only for a specific period. Such measure can also be useful in network monitoring, stock market, etc.

Therefore, in this paper, we address the problem of discovering *regular* patterns in a transactional database. We define a new *regularity* measure for a pattern by the maximum interval of its consecutive occurrences in the whole database. Thus, *regular* patterns, defined such way, satisfy the downward closure property [1]. We propose a novel tree structure, called RP-tree (Regular Pattern tree), to capture the database contents in a highly compact manner and mine *regular* patterns from it by using an efficient pattern growth-based mining technique. Our extensive performance study shows that mining *regular* patterns from RP-tree is highly memory and time efficient.

The rest of the paper is organized as follows. Sections 2 summarizes the existing algorithms to mine *periodic* and *cyclic* patterns that are mostly related to our work. Section 3 introduces the problem definition of *regular* pattern mining. The structure of RP-tree and *regular* pattern mining technique are given in Section 4. We report our experimental results in Section 5. Finally, Section 6 concludes the paper.

2 Related Work

Mining frequent patterns [2, 7, 4], *periodic* patterns [8, 6, 9] and *cyclic* patterns [3] in static database have been well-addressed over the last decade. Han et. al. [2] proposed the frequent pattern tree (FP-tree) and the FP-growth algorithm to mine frequent patterns with a memory and time efficient manner. The FP-growth's performance gain is mainly based on the highly compact support-descending FP-tree structure.

Periodic pattern mining problem in time-series data focuses on the cyclic behavior of patterns either in the whole [8] or at some point [9] of time-series. Such pattern mining has also been studied as a wing of sequential pattern mining [9, 6] in recent years. However, although *periodic* pattern mining is closely related with our work, it cannot be directly applied for finding *regular* patterns from a transactional database because it considers either time-series or sequential data.

Ozden et. al. [3] proposed a method to discover the association rules [1] occurring cyclically in a transactional database. It outputs a set of rules that maintains a *cyclic* behavior in appearance among a predefined non-overlapping database segments. The main limitation of this method is segmenting the database into a series of fixed sized segments, which may suffer from “border effect”. That is, if the sufficient number of occurrences of a pattern (to become frequent) occurs in the borders of two consecutive segments, the pattern might be ignored to generate association rules.

3 Problem Definition

Let $L = \{i_1, i_2, \dots, i_n\}$ be a set of items. A set $X \subseteq L$ is called a *pattern* (or an *itemset*). A transaction $t = (tid, Y)$ is a couple where *tid* is the transaction-id and Y is a pattern. A transactional database DB is a set of transactions $T = \{t_1, \dots, t_m\}$ with $m = |DB|$, i.e., total number of transactions in DB . If $X \subseteq Y$, it is said that X occurs in t and denoted as t_j^X , $j \in [1, m]$. Thus, $T^X = \{t_j^X, \dots, t_k^X\}$, $j \leq k$ and $j, k \in [1, m]$ is the set of all transactions where pattern X occurs. Let t_{j+1}^X and t_j^X , are two consecutive transactions in T^X . Then $p^X = t_{j+1}^X - t_j^X$, $j \in [1, (m-1)]$ is a period of X and $P^X = \{p_1^X, \dots, p_r^X\}$ is the set of all periods of X in DB . For simplicity, we consider the first and the last transactions in DB as ‘null’ with $t_{first} = 0$ and $t_{last} = t_m$ respectively. Let the *max_period* of $X = \text{Max}(t_{j+1}^X - t_j^X)$, $j \in [1, (m-1)]$ be the largest period in P^X . We take *max_period* as the *regularity* measure for a pattern and denote as $reg(X)$ for X .

Therefore, a pattern is called a *regular* pattern if its *regularity* is no more than a user-given maximum *regularity* threshold called *max_reg* λ , with $1 \leq \lambda \leq |DB|$. *Regular pattern mining problem*, given a λ and a DB , is to discover the complete set of *regular* patterns having *regularity* no more than λ in the DB .

4 RP-Tree: Design, Construction and Mining

Since *regular* patterns follow the downward closure property, with one DB scan we identify the set of length-1 *regular* items say, R for a given *max_reg*. An item header table, called *regular* table (R-table in short), is built with this scan in order to facilitate the tree traversal and to store all length-1 items with respective *regularity* and support. Each entry in R-table consists of four fields in sequence (i, s, t_i, r); item name (i), support (s), *tid* of the last transaction where i occurred (t_i), and the *regularity* of i (r). Let t_{cur} and p_{cur} be the *tid* of current transaction and the most recent period respectively for an item X . The R-table is, therefore, maintained according to the process given in Fig. 1. The first transaction ($t_{cur} = 1$), $\{ad\}$ initializes all entries for item ‘a’ and ‘d’ in R-table, as shown in Fig. 2(a). The next transaction ($t_{cur} = 2$) sets R-table entries for items ‘b’, ‘c’ and ‘e’ with the values $\{s; t_i; r\} = \{1; 2; 2\}$ and updates the same for ‘a’ (Fig. 2(b)). The R-table, after scanning up to $tid = 9$, is given in Fig. 2(c). To reflect the correct period for each item, the whole table is refreshed by updating r values (considering $t_{cur} = 9$) of each item at the end of DB as shown in Fig. 2(d). Once the R-table is built, we generate R by removing all *irregular* items and arranging the items in support-descending order to facilitate the RP-tree construction.

1. **If** t_{cur} is X 's first occurrence
2. $\{s = 1, t_l = t_{cur}, r = t_{cur}\};$
3. **Else** $\{s = s + 1;$
4. $p_{cur} = t_{cur} - t_l, t_l = t_{cur};$
5. **If** $(p_{cur} > r)$
6. $r = p_{cur}; \}$
7. Refresh the table at the end of $DB;$

R-table
i, s, t_l, r
a:1;1;1
b:
c:
d:1;1;1
e:
f:

R-table
i, s, t_l, r
a:2;2;1
b:1;2;2
c:1;2;2
d:1;1;1
e:1;2;2
f:

R-table
i, s, t_l, r
a:5;5;1
b:6;9;3
c:5;9;2
d:5;9;5
e:6;8;2
f:3;8;3

R-table
i, s, t_l, r
a:5;5;4
b:6;9;3
c:5;9;2
d:5;9;5
e:6;8;2
f:3;8;3

(a) After $tid = 1$ (b) After $tid = 2$ (c) After $tid = 9$ (d) After refreshing

Fig. 1. R-table maintenance

Fig. 2. R-table population for the DB in Table-1

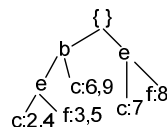
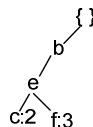
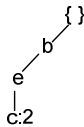
4.1 Construction of an RP-Tree

With the second DB scan, using the FP-tree construction technique [2], the RP-tree is constructed in such a way that, it only contains items in R in R-table order. No node in an RP-tree does maintain the support count field. However, the transaction occurrence information is explicitly kept in a list called tid -list in the last node (say, $tail$ -node) of the transaction. The following example illustrates the construction of an RP-tree. Consider the DB of Table 1, and Fig. 3 for the step-by-step RP-tree construction for $\lambda = 3$. Figure 3(a) shows the R-table (for $\lambda = 3$) obtained from the R-table of Fig. 2(d). For the simplicity of figures, we do not show the node traversal pointers in trees, however, they are maintained in a fashion like FP-tree does.

Since all the items in $tid = 1$ are *irregular*, the first transaction to be inserted is $\{abce\}$ (i.e., $tid = 2$). After removing *irregular* item(s) from $tid = 2$ and sorting the *regular* items, we insert $\{abce\}$ in the form and order of $\{bec\}$ in tree with node “c:2” being the $tail$ -node that carries the tid for the transaction, as shown in Fig. 3(b). For the next transaction (i.e., $tid = 3$), as in Fig. 3(c), since its (ordered) *regular* item list (b, e, f) shares a common prefix (b, e) with the existing path (b, e, c) , one new node (“f:3”) is created as a $tail$ -node with value 3 in its tid -list and linked as a child of node (“e”). After scanning all the transactions and inserting them in similar fashion, the final RP-tree for the DB of Table 1 with $\lambda = 3$ is shown in Fig. 3(d). Based on above R-table and RP-tree construction techniques, we have the following property and lemma of an RP-tree. For each transaction t in DB , $reg(t)$ is the set of all *regular* items in t , i.e., $reg(t) = item(t) \cap R$, and is called the *regular* item projection of t .

Property 1. An RP-tree maintains a complete set of *regular* item projection for each transaction in a DB only once.

R-table
i, s, t_l, r
b:6;9;3
e:6;8;2
c:5;9;2
f:3;8;3



(a) Regular items ($\lambda = 3$) (b) After inserting $tid = 2$ (c) After inserting $tid = 3$ (d) After inserting $tid = 9$

Fig. 3. Construction of an RP-tree

Lemma 1. Given a transactional database DB and a max_reg , the complete set of all *regular* item projections of all transactions in DB can be derived from the RP-tree for the max_reg .

Proof. Based on the RP-tree construction mechanism, $reg(t)$ of each transaction t is mapped to only one path in it and any path from the *root* up to a *tail*-node maintains the complete projection for exactly n transactions (where n is the total number of entries in the *tid*-list of the *tail*-node). ■

Based on the RP-tree construction process, Property 1 and Lemma 1, each transaction t contributes at best one path of the size $|reg(t)|$ to an RP-tree. Therefore, the total size contribution of all transactions can be $\sum_{t \in DB} |reg(t)|$ at best. However, since there is usually a lot of common prefix patterns among the transactions, the size of an RP-tree is normally much smaller than $\sum_{t \in DB} |reg(t)|$. One may assume that the structure of the RP-tree may not be memory efficient, since it explicitly maintains all *tids* in the tree structure. However, we argue that the RP-tree achieves the memory efficiency by keeping such transaction information only at the *tail*-nodes and avoiding the support count field at each node. Moreover, keeping the *tid* information in tree structure has also been found in literature for efficiently mining frequent patterns [5], [7]. To certain extent, some of those works additionally maintain support count and/or the *tid* information [7] in each tree node.

4.2 Mining with RP-Tree

The support-descending RP-tree, constructed in above example, enables the subsequent mining of *regular* patterns with a rather compact data structure. Similar to FP-growth-based [2] mining approach, we recursively mine the RP-tree of decreasing size to generate *regular* patterns by creating pattern-bases and corresponding conditional trees without additional DB scan. Before discussing the mining process we explore the following important property and lemma of an RP-tree.

Property 2. Each *tail*-node in an RP-tree maintains the occurrence information of all the nodes in the path (from that *tail*-node to *root*) in the transactions of its *tid*-list.

Lemma 2. Let $Z = \{a_1, a_2, \dots, a_n\}$ be a path in an RP-tree where node a_n , being the *tail*-node, carries the *tid*-list of the path. If the *tid*-list is pushed-up to node a_{n-1} , then node a_{n-1} maintains the occurrence information of path $Z' = \{a_1, a_2, \dots, a_{n-1}\}$ for the same set of transactions in the *tid*-list without any loss.

Proof. Based on Property 2, the *tid*-list in node a_n explicitly maintains the information about the occurrence of Z' for the same set of transactions. Therefore, the same *tid*-list at node a_{n-1} exactly maintains the same information for Z' without any lose. ■

The pattern-base is constructed for each item starting from the bottom of the R-table. The pattern-base for an item i , PB_i is created by accumulating only the prefix sub-paths

of nodes labeled i in the RP-tree. Since i is the last item in the R-table, each node labeled i in the RP-tree must be a *tail*-node. Therefore, based on Lemma 2, the *tid*-lists of all such *tail*-nodes are pushed-up to respective parent nodes in the RP-tree and in PB_i . Thus, the parent node is converted to a *tail*-node if it was an *ordinary* node; otherwise, the *tid*-list is merged with its previous *tid*-list. All nodes labeled i in the RP-tree and the entry for i in R-table are, thereafter, deleted. The pattern-base for ' f ' of Fig. 3(d) is shown in Fig. 4(a). The *tid*-list of every *tail*-node of any PB_i is mapped (in temporary array) to all items in the respective path to compute the *regularity* of each item j in R-table _{i} . Therefore, it is rather simple calculation to compute $reg(ij)$ from T^{ij} by generating P^{ij} , as shown in Fig. 4(b) for the PB_f . The conditional tree for i CT_i is, then, constructed from PB_i by removing all *irregular* items and nodes respectively from R-table _{i} and PB_i . The *tid*-list (if any) of the deleted node is pushed-up to its parent node. The conditional tree for ' f ' can be generated as shown in Fig. 4(c). The whole process of pattern-base and conditional tree constructions is repeated until the R-table becomes empty.

From the above mining process the complete set of *regular* patterns for a given max_reg can be generated from an RP-tree constructed on a DB . The technique is efficient due to support-descending item order and performing the mining operation from bottom to top.

5 Experimental Results

We performed comprehensive experimental analysis on the performance of RP-tree to discover *regular* patterns over several synthetic and real datasets which are frequently used in frequent pattern mining experiments, since such datasets maintain the characteristics of transactional database. Due to the space constraint we only report the results on a subset of them. All programs are written in Microsoft Visual C++ 6.0 and run with Windows XP on a 2.66 GHz machine with 1GB of main memory. Runtime specifies the total of CPU and I/Os times.

The memory consumptions of the RP-tree for different values of max_reg over dense real dataset *chess* and sparse synthetic dataset *T10I4D100K* are reported in Fig. 5. The more the value of max_reg is, the more the memory RP-tree requires. The reason is that, with the increase of max_reg the number of *regular* patterns increases for every dataset, as shown in Table 2. Therefore, the size of RP-tree becomes larger to store more patterns. However, it is clear from the figure that, the structure of an RP-tree can easily be handled in a memory efficient manner.

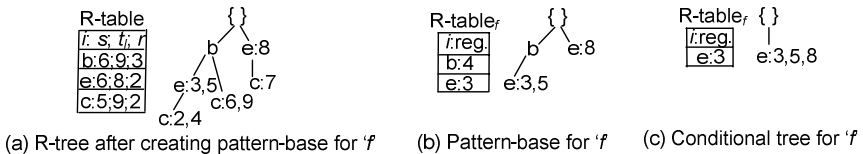


Fig. 4. Pattern-base and Conditional tree construction with the RP-tree

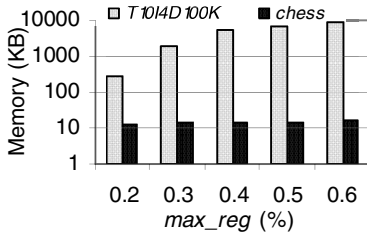


Fig. 5. Compactness of the RP-tree

Table 2. Pattern count on max_reg

Dataset	max_reg (%)	Number of Patterns
T10I4D100K	0.2	19
	0.6	309
chess	0.1	5
	0.6	4839

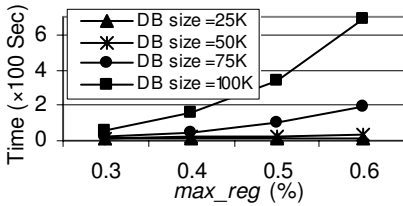


Fig. 6. Execution time over T10I4D100K

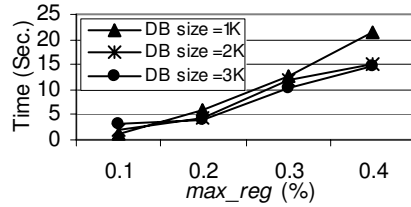


Fig. 7. Execution time over chess

We observed the execution time the RP-tree requires in mining regular patterns on change of max_reg. The execution time includes the construction of the R-table and RP-tree, and corresponding mining. The execution time may vary depending on the size of dataset keeping the characteristic fixed. Therefore, to grasp the effect of mining on variation of both max_reg and dataset size, we perform regular pattern mining by increasing the size of T10I4D100K (Fig. 6) from 25K to 100K (full dataset) and that of chess (Fig. 7) from 1K to 3K (full dataset) with the variation of max_reg. The results show that, as per as the partial or whole DB and reasonably high max_reg are concerned, mining regular patterns from the corresponding RP-tree is rather time efficient for both sparse and dense datasets.

We also study the scalability of the RP-tree by varying the number of transactions in DB. We use real Kosarak dataset for the scalability experiment, since it is a huge sparse dataset with large number of distinct items (41,270) and transactions (990,002). We divided the dataset into five portions of 0.2 million transactions in each part. Then we investigated the performance of the RP-tree after accumulating each portion with previous parts with performing regular pattern mining each time. We fixed the max_reg with 0.1%. As shown in Table 3, as the DB increases, the execution time and required memory increase. However, RP-tree shows stable performance of about linear increase of runtime and memory consumption with respect to the size of DB.

Table 3. Scalability of the RP-tree

Dataset: Kosarak max_reg = 0.1%	Dataset Size (Million Transactions)				
	0.2	0.4	0.6	0.8	1.0
Execution Time (Sec)	57.70	126.23	224.27	364.60	635.20
Memory (MB)	0.74	1.72	4.23	6.64	9.78

6 Conclusions

We have introduced a new interesting pattern mining problem, called *regular* patterns, that explores the temporal regularity of pattern occurrence in transactional database. To efficiently mine *regular* patterns, we have also proposed a highly compact tree structure called the RP-tree. The experimental results demonstrate that our RP-tree can provide the time and memory efficiency during the *regular* pattern mining. Moreover, it is highly scalable in terms of time and memory.

References

1. Agrawal, R., Imielinski, T., Swami, A.N.: Mining Association Rules Between Sets of Items in Large Databases. In: ACM SIGMOD International Conference on Management of Data, pp. 207–216 (1993)
2. Han, J., Pei, J., Yin, Y.: Mining Frequent Patterns without Candidate Generation. In: ACM SIGMOD International Conference on Management of Data, pp. 1–12. (2000)
3. Ozden, B., Ramaswamy, S., Silberschatz, A.: Cyclic Association Rules. In: 14th International Conference on Data Engineering, pp. 412–421 (1998)
4. Tanbeer, S.K., Ahmed, C.F., Jeong, B.-S., Lee, Y.-K.: CP-tree: A Tree Structure for Single-Pass Frequent Pattern Mining. In: PAKDD (accepted to be published, 2008)
5. Chi, Y., Wang, H., Yu, P.S., Muntz, R.R.: Catch the Moment: Maintaining Closed Frequent Itemsets Over a Data Stream Sliding Window. *Knowledge and Information System* 10(3), 265–294 (2006)
6. Maqbool, F., Bashir, S., Baig, A.R.: E-MAP: Efficiently Mining Asynchronous Periodic Patterns. *International Journal of Computer Science and Network Security* 6(8A), 174–179 (2006)
7. Zhi-Jun, X., Hong, C., Li, C.: An Efficient Algorithm for Frequent Itemset Mining on Data Streams. In: Perner, P. (ed.) *International Conference on Management of Data*, pp. 474–491 (2006)
8. Elfeky, M.G., Aref, W.G., Elmagarmid, A.K.: Periodicity Detection in Time Series Databases. *IEEE Transactions on Knowledge and Data Engineering* 17(7), 875–887 (2005)
9. Lee, G., Yang, W., Lee, J.-M.: A Parallel Algorithm for Mining Multiple Partial Periodic Patterns. *Information Sciences* 176, 3591–3609 (2006)

Extracting Key Entities and Significant Events from Online Daily News

Mingrong Liu, Yicen Liu, Liang Xiang, Xing Chen, and Qing Yang

National Laboratory of Pattern Recognition
Institute of Automation, Chinese Academy of Sciences
95 Zhongguancun Donglu, Beijing 100190, China
{mrliu,ycliu,lxiang,xchen,qyang}@nlpr.ia.ac.cn

Abstract. To help people obtain the most important information daily in the shortest time, a novel framework is presented for simultaneous key entities extraction and significant events mining from daily web news. The technique is mainly based on modeling entities and news documents as weighted undirected bipartite graph, which consists of three steps. First, key entities are extracted by scoring all candidate entities on a specific day and tracking their trends within a specific time window. Second, a weighted undirected bipartite graph is built based on entities and related news documents, then mutual reinforcement is imposed on the bipartite graph to rank both of them. Third, clustering on news articles generates daily significant events. Experimental study shows effectiveness of this approach.

Keywords: web news mining, entity, mutual reinforcement, event.

1 Introduction

Every day there is a huge amount of new information available to us, and a large portion of it is news on the web. Online newspapers and news portals have become one of the most important sources of up-to-date information. However, without proper organization of the overwhelming information, one can easily become lost because of its vast size. It is not feasible for a web surfer to go through all the news without any pre-processing, because the news a person can read is much less than the amount produced within the same time period on the web. Thus, there is a growing need for tools that will allow individuals to access and keep track of this information in a automatic manner. To help people obtain the most important information daily in the shortest time, a system should be designed to automatically extract significant ones from web news repository.

In this paper, we are interested in extracting key entities and significant events from daily text-based web news documents. Our aim is to alleviate the information overload problem by focusing on important events, that is, events that are popular during a specified time period and typically contain several related key entities that are the basis of events extraction. We introduce two critical properties of a key entity, “novelty” and “pervasiveness”. For “novelty”, we mean that the relative importance of an entity is in an ascending trend during the most

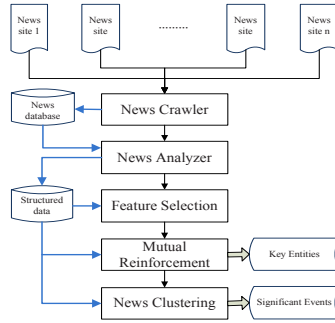


Fig. 1. System Architecture

recent days, for example, one week in our experiment; by “pervasiveness” we evaluate its popularity on current day. For example, the entity “Buenos Aires” (Argentine Capital) has both properties on April 12th, 2008, since “Buenos Aires” was seldom reported (in our news repository) during the past week while widely mentioned on that day, significant event related to it is that Olympic flame arrives in Buenos Aires for the first time. Figure 1 shows the system architecture of the proposed framework. News documents are downloaded from several news sites on the web by crawling those sites with reasonable frequency every day. Fetched documents are firstly stored in local databases, then analyzed using web news parsing tools to build specific data structure of entities and documents for further computation. Novel entities among all candidate ones are extracted as features by taking their “novelty” properties into account. A bipartite graph is built based on extracted features and documents related to them. Through mutual reinforcement between features and documents, we can simultaneously rank both of them. By considering both ranking score and pre-computed “novelty” value, key entities are extracted on the day. Clustering are performed on ranked documents, and the ranking score of a cluster is the sum of its contained elements. Clusters with highest scores are most significant events on each day. In our implemented online system, multi-document summarization of each cluster is provided to help user quickly get main knowledge of that event. In addition to the list of hot entities, one can also see the trends and related news of them during last week. To the best of our knowledge, this is a novel framework on the subject.

The rest of this paper is organized as follows. Section 2 briefly reviews the related work. Section 3 deals with the feature selection issue. Sections 4 and 5 present the proposed extracting method. We report the experimental results in Section 6 and conclude in Section 7.

2 Related Work

Our work is related to several lines of work in text mining and stream data mining. Topic detection and tracking (TDT) is the major area that tackles the

problem of discovering events from the news articles. Most of them focused on online detection [2,3,4]. Our proposed framework is also based on online news, however, we intend to extract key entities and significant events while TDT mainly aims to detect new events.

Capturing variations in the distribution of key terms on a time line is a critical step in extracting significant events. It is essential to track the terms to determine what stage of their life cycle they are in [5]. Bursty patterns or events are recently studied. [8,9] proposed using χ^2 -test to construct an overview timeline for the features in the text corpus. In this paper, key entities are also extracted as features. In addition, our system can simultaneously extract significant events, which are highly related to those features. [6] proposed an algorithm for constructing a hierarchical structure for the features in the text corpus by using an infinite-state automaton. [14] constructs a feature-based event hierarchy for a news text corpus, which is mainly based on clustering news documents to form events and organize them in a hierarchical structure.

There are relatively fewer work which aims to extract significant events. [11] extracts hot topics from a given set of text-based news documents published during a given time period, the goal is similar to our work, however, we focus on hot events on current day, not in past days.

3 Extracting Entities as Features

For news articles, entities are main feature terms, especially named entities (*People, Locations, Organizations, etc.*), so we treat them as features. In this section, we focus on extract “novel” entities. Algorithm 1 shows how candidate entities

Algorithm 1. Feature Selection

Require: Entity dict and news data repository.

Ensure: Top n features.

$FS = \phi$

for all entity term t such that t appears in more than two documents **do**

$val_{curday} = tf/idf$ value of t

if $val_{curday} > \theta$ **then**

track its tf/idf value in each of past w days

compute its novelty value using Minimum Least Square

add pair (t, val_{curday}) into FS

end if

end for

sort FS according to novelty value of entities

return top n terms as features

are selected as features. For each candidate entity, its significance s , on current day is computed, if s is above a given threshold θ , its significance is computed for each day in past n days. Then we track its trend by the method of curve

fitting, such as the simple and effective least square method. The slope of fitted line k is taken as novelty value of the corresponding entity. The top entities, such as top 500 in this paper, are extracted as features for further computation. Entity significance s on a specific day is computed through traditional tf/idf evaluation method in the information retrieval literature. tf is term frequency of an entity on a specific day. Since our consideration is news article, news title is more important than text, a bonus is added to its tf if an entity appears in news titles. idf of t is the inverse document frequency of term t , which is usually computed as $idf(t) = \log \frac{N-n(t)+0.5}{n(t)+0.5}$, where N is the number of documents on the specific day, and $n(t)$ is the number of documents in which term t occurs.

4 Mutual Reinforcement

In previous step, novel entities are selected as feature terms $T = \{t_1, \dots, t_m\}$. Using traditional inverted indexing structure, all related documents set can be listed as $D = \{d_1, \dots, d_n\}$, each of D contains at least one feature. Taking elements in T and D as vertices, a bipartite graph can be built from T and D in the following way: if the feature term t_i has relation with document d_j , then create an edge between t_i and d_j . We specify positive weights on the edges of the bipartite graph with w_{ij} which indicates the weight on the edge (t_i, d_j) . w_{ij} is calculated using the probabilistic model, Okapi BM25: $score(d, t) = idf(t) \cdot \frac{f(t, d) \cdot (k+1)}{f(t, d) + k \cdot (1-b) + b \cdot \frac{|d|}{avgdl}}$

where $f(t, d)$ is t 's term frequency in the document d , $|d|$ is the length of the document d (number of terms), and $avgdl$ is the average document length in the text collection from which documents are drawn. k and b are free parameters, usually chosen as $k = 2.0$ and $b = 0.75$ empirically.

BM25 is a ranking function used by search engines to rank matching documents according to their relevance to a given search query, representing state-of-the-art retrieval functions used in document retrieval. We denote the weighted bipartite graph by $G(T, D, W)$, where $W = [w_{ij}]$ is the m -by- n weight matrix containing all the pairwise edge weights. For each feature t_i and each document d_j we wish to compute their saliency scores $u(t_i)$ and $v(d_j)$, respectively. To this end, we state the following mutual reinforcement principle:

A term should have a high saliency score if it appears in many documents with high saliency scores while a document should have a high saliency score if it contains many terms with high saliency scores.

The idea is similar to web page ranking method used to find the hub and authority pages in a link graph [13]. Essentially, the principle indicates that the saliency score of a term is determined by the saliency scores of the documents it appears in, and the saliency score of a document is determined by the saliency scores of the terms it contains. However, unlike web page ranking which totally depends on the structure of link graph between web pages, since we have prior information of terms, the ‘‘novelty value’’, we add this information to the mutual reinforcement principle. Mathematically, the above statement is rendered as:

$$u(t_i) \propto \sum_{v(d_j) \sim u(t_i)} nov_{t_i} w_{ij} v(d_j), \quad v(d_j) \propto \sum_{u(t_i) \sim v(d_j)} nov_{t_i} w_{ij} u(t_i)$$

where the summations are over the neighbors of the vertices in question, and $a \sim b$ indicates there is an edge between vertices a and b , i.e., when computing a term score, the summation is over all documents that contain the term and when computing a document score, the summation is over all terms that appear in the document. Note that the prior information of novelty value is taken into account. The symbol \propto stands for proportional to. Now we collect the saliency scores for terms and documents into two vectors u and v , respectively, the above equation can then be written in the following matrix format $u = \frac{1}{\sigma}DWv$, $v = \frac{1}{\sigma}W^TDu$, where D is a diagnosis matrix where D_{ii} is the pre-computed novelty value of each corresponding feature. W is the weight matrix of the bipartite graph of features and documents, W^T stands for the matrix transpose of W , and $\frac{1}{\sigma}$ is the proportionality constant. It is easy to see that u and v are the left and right singular vectors of DW corresponding to the singular value σ . If we choose σ to be the largest singular value of DW , then it is guaranteed that both u and v have nonnegative components. The corresponding component values of u and v give the feature and document saliency scores, respectively.

For the numerical computation of the largest singular value triplet u, σ, v , we can use a variation of the power method adapted to the case of singular value triplets: choose an initial value for v to be the vector of all ones. Iterate the following two steps until convergence, (1) $u = DWv, u = u/\|u\|$, (2) $v = W^TDu, v = v/\|v\|$, where the vector norm $\|\cdot\|$ can be chosen to the Euclidean norm, and $\sigma = u^TDWv$ upon convergence. For a detailed analysis of the singular value decomposition for related types of matrices, the reader is referred to [10].

Algorithm 2. Clustering news documents into events

Require: News documents.

Ensure: News Events.

1. assign each news document to a cluster. Let the similarities between the clusters be the same as the similarities between the items they contain.
2. find the closest (most similar) pair of clusters and merge them into a single cluster.
3. compute similarities between the new cluster and each of the old clusters.

Cluster similarity between X and Y : $Sim(X, Y) = \frac{1}{N_X \times N_Y} \sum_{x \in X, y \in Y} sim(x, y)$

4. Repeat steps 2 and 3 until similarity between closest pair is below an threshold.
 5. Return each cluster as an event.
-

Now both rankings of entities and news documents are calculated. Top n entities are selected as key entities on current day. Next clustering of news documents will generate significant events.

5 Clustering Documents to Generate Significant Events

In this section, we cluster news documents into groups to generate significant events. Hierarchical Aggregative Clustering(HAC) is chosen to cluster news documents. Algorithm 2 shows the method. Note that in algorithm 2, N_X and N_Y

are number of news documents in cluster X and Y respectively. Only news titles are taken into account to compute the similarity of two news documents x and y : Extract named entities NE and other entities OE in each document title, then $sim(x, y)$ is computed as $sim(x, y) = \alpha sim(NE_x, NE_y) + \beta sim(OE_x, OE_y) + \gamma sim(Title_x, Title_y)$. $sim(NE_x, NE_y)$ et. can be calculated using string similarity measurement. The sum of α , β , and γ is 1 and they are chosen empirically, in this paper they are set to 0.5, 0.3 and 0.2 respectively.

The ranking of each cluster (representing an event) is the sum of ranking of its contained news documents. We select top n (say 50) clusters as significant events. In the online system¹, for each significant event, summary is generated using multi-document summarization and related pictures in those news are provided (if exists) to help user quickly acquire an overview of that event. Due to space size and the aim of this paper, we will not discuss detail of those work.

6 Experimental Study

6.1 Data Sets

About 20 news sites are selected in our current online system, and all of them are important and popular Chinese web news sites. News sites are crawled every 2 to 3 hours, and the number of news documents fetched everyday is around 3000 in average. They are general news ranging from politics, economics, societies to sports and entertainment, etc., and most of them are related to China. A news document parsing tool is developed to automatically extract news texts from the original html sources. We utilize Chinese words segmenting tool with part-of-speech tagging to extract candidate entities, then structured data is built and stored in local databases.

6.2 Key Entities

Figure 2 depicts trends of the most novel entities on April 12th, and May 10th, 2008. Note that they are not the final key entities. Table 3 illustrates the extracted key entities on April 12th, 2008². We will see that most of them are related to significant events on that day. Most of the extracted entities are named entities. Entities 2, 6, 13 are people names, entities 1, 7, 8, 9, 11, 14, 16, 17, 18 are location names and entities 3, 4, 5, 10, 12, 15 are general entities.

6.3 Significant Events

Table 4 lists extracted significant events on April 12th, 2008. By reading online news on several news sites, we know that there are several significant topics on that day: Boao Forum for Asia 2008, which is an important forum for Asia on economics; Beijing Olympic torch; Poverty line in China, etc. We can see in table 4

¹ URL: <http://v.cindoo.com/news/newsminer.html>

² We have translated the experimental results into English in this paper.

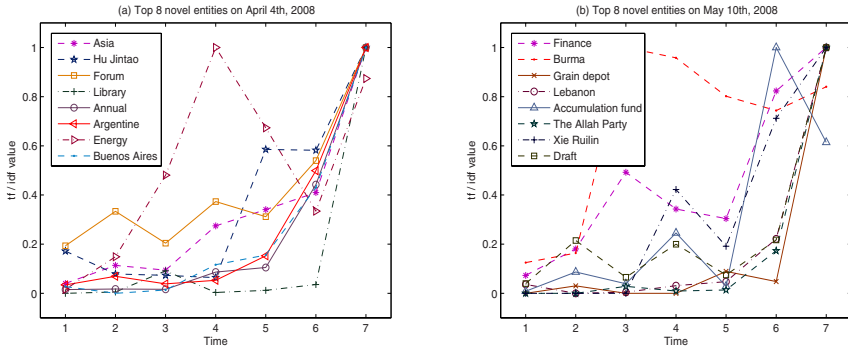


Fig. 2. Top features with high novelty score. (a): Asia, Hu Jintao, Forum, Library, Annual, Argentine, Energy, Buenos Aires, on April 12th, 2008 and (b): Finance, Burma, Grain depot, Lebanon, Allah Party, Xie Ruilin, Draft, Accumulation fund, on May 10th, 2008. Note that on the time axis, 7 represents current day, 6-1 represent pre 6 days, and the tf/idf values are normalized in [0,1].

Table 3. Top 18 Key Entities on April 12th,2008

Asia (0.263602)	Hu Jintao (0.229216)	Forum(0.160884)
Annual (0.111732)	Energy (0.0319571)	Long Yongtu (0.011728)
Chile (0.0091095)	Kazakhstan (0.00847223)	Tanzania (0.0070724)
Climate (0.00653906)	Buenos Aires (0.00551221)	Theme (0.0054622)
Wright (0.00535269)	Mongolia (0.00443333)	Heads of State(0.0028912)
Sanya (0.00245002)	Qatar (0.00232734)	Hainan Province (0.0022878)

Table 4. Top 10 events on April 12th,2008

No.	Event
1	Boao Forum for Asia 2008 opens.
2	A dinner hosted by Chinese President Hu Jintao honoring participants at Boao.
3	Hu Jintao meets Xiao Wanchang.
4	CNOOC chief: Energy prices unlikely to soar in short term.
5	European Parliament debates human rights in China.
6	Poverty line to be raised to 1,300 yuan(\$186) in China.
7	Scientists predict: North Earth will be moist while arid in the south in 10 years.
8	Premier of Malaysia in charge of election defeat and will transfer the power.
9	Chinese Ambassador to Japan went to Nagano preparing for Olympic torch relay.
10	Beijing Olympic torch passing in Buenos Aires.

that events 1,2,3 are related to Boao Forum for Asia; Event 6 is about poverty line in China; Events 9 and 10 are related to Olympic torch. Event 4 talks about energy problem, event 5 is about human rights. Events 7 and 8 focus on global climate and significant political event in Malaysia. It's clear that all those 10 events are significant general events, which are in areas of politic, economic,

sports, and social life. Due to space limit, we will not list more results in this paper, the reader can browse our online system for more recent results.

7 Conclusions and Future Work

In this paper, we have proposed a framework for extracting key entities and significant events from online news articles that appear in news sites. Our work makes novel and important contributions in the following aspects: by tracking history of entities, novel entities are extracted on current day, then a bipartite graph is built based on those entities and news documents related to them, and mutual reinforcement principle is imposed on the graph to simultaneously rank entities and news documents. Experimental study shows effectiveness and efficiency of our proposed work, which can help people quickly get knowledge of key entities and significant events on each day. As future work, we are planning to expand our experiment in two directions. First, we will conduct user studies to further measure the effectiveness of our proposed framework. Second, we intend to apply the proposed framework to domain-specific news, such as entertainment news, sports news and finance news. More precise results are expected.

References

1. Li, W., Qian, D., Lu, Q., Yuan, C.: Detecting, categorizing and clustering entity mentions in Chinese text. In: SIGIR 2007, pp. 647–654 (2007)
2. Allan, J., Papka, R., Lavrenko, V.: On-line new event detection and tracking. In: Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 1998) (1998)
3. Connell, M., Feng, A., Kumaran, G., Raghavan, H., Shah, C., Allan, J.: UMass at TDT 2004. In: 2004 Topic Detection and Tracking Workshop (TDT 2004), Gaithersburg, Maryland, USA (2004)
4. Allan, J., Carbonell, J., Doddington, G., Yamron, J., Yang, Y.: Topic detection and tracking pilot study: Final report. In: Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop (1998)
5. Chen, C.C., Chen, Y.T., Sun, Y., Chen, M.C.: Life cycle modeling of news events using aging theory. In: Lavrač, N., Gamberger, D., Todorovski, L., Blockeel, H. (eds.) ECML 2003. LNCS (LNAI), vol. 2837, pp. 47–59. Springer, Heidelberg (2003)
6. Kleinberg, J.M.: Bursty and hierarchical structure in streams. In: Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 91–101 (2002)
7. Brants, T., Chen, F.: A system for new event detection. In: Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2003) (2003)
8. Smith, D.A.: Detecting and browsing events in unstructured text. In: Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2002) (2002)
9. Swan, R.C., Allan, J.: Extracting significant time varying features from text. In: Proceedings of the 1999 ACM CIKM International Conference on Information and Knowledge Management (CIKM 1999) (1999)

10. Zha, H., Zhang, Z.: On Matrices with Low-rank-plus-shift Structures: Partial SVD and Latent Semantic Indexing. *SIAM Journal of Matrix Analysis and Applications* 21, 522–536 (1999)
11. Chen, K.-Y., Luesukprasert, L., Chou, S.-c.T.: Hot topic extraction based on timeline analysis and multidimensional sentence modeling. *IEEE transactions on knowledge and data engineering* 19(8) (August 2007)
12. Fung, G.P.C., Yu, J.X., Yu, P.S., Lu, H.: Parameter free bursty events detection in text streams. In: *VLDB*, pp. 181–192 (2005)
13. Kleinberg, J.: Authoritative sources in a hyperlinked environment. In: *Proceedings of the 9th ACM-SIAM Symposium on Discrete Algorithms* (1998)
14. Fung, G.P.C., Yu, J.X., Liu, H., Yu, P.S.: Time-dependent event hierarchy construction. In: *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2007)
15. Yang, Y., Pierce, T., Carbonell, J.: A study of retrospective and on-line event detection. In: *SIGIR*, pp. 28–36 (1998)

Performance Evaluation of Intelligent Prediction Models on Smokers' Quitting Behaviour

Chang-Joo Yun¹, Xiaojiang Ding¹, Susan Bedingfield¹, Chung-Hsing Yeh¹,
Ron Borland², David Young², Sonja Petrovic-Lazarevic³, Ken Coghill³,
and Jian Ying Zhang³

¹ Clayton School of Information Technology, Monash University, Victoria, Australia

² VicHealth Centre for Tobacco Control, the Cancer Council Victoria, Australia

³ Department of Management, Monash University, Victoria, Australia

Abstract. This paper evaluates the performance of intelligent models using decision trees, rough sets, and neural networks for predicting smokers' quitting behaviour. 18 models are developed based on 6 data sets created from the International Tobacco Control Four Country Survey. 13 attributes about smokers' beliefs about quitting (BQ) and 13 attributes about smokers' beliefs about smoking (BS) are used as inputs. The output attribute is the smokers' status of making a quit attempt (MQA) or planning to quit (PTQ). The neural network models outperform both decision tree models and rough set models in terms of prediction ability. Models using both BQ and BS attributes as inputs perform better than models using only BQ or BS attributes. The BS attributes contribute more to MQA, whereas the BQ attributes have more impact on PTQ. Models for predicting PTQ outperform models for predicting MQA. Determinant attributes that affect smokers' quitting behaviour are identified.

Keywords: Decision trees, rough sets, neural networks, tobacco control.

1 Introduction

Smoking has long been a recognized global health problem. Various countries have attempted to influence the behaviour of smokers using means such as warning labels on cigarette packs, increasing the price of cigarettes and TV advertisements. The International Tobacco Control (ITC) Policy Evaluation Project aims at investigating the psychosocial and behavioural effects of tobacco control policies by conducting studies based on survey data from the ITC Four Country Survey (ITC-4C). The ITC project involves surveying on adult smokers from the United States, Canada, the United Kingdom and Australia [1]. Empirical evidence based on ITC-4C has shown that smokers' internal motivations and beliefs about quitting have been identified as key factors to quitting [2]. Tobacco control policies have been implemented by governments in all the countries mentioned. The intention of these policies is to encourage smokers to quit smoking.

Among the key research issues to be addressed by the ITC, predicting the response of smokers to tobacco control policies has been a significant and challenging research task. Existing studies using statistical techniques focus on examining the relationship

between government policies and smokers' behaviours [3][4][5]. However, these techniques have limitations in dealing with complex non-linear data.

To address this important issue, we develop intelligent prediction models using decision trees (DT), rough sets (RS), and neural networks (NN) to predict smokers' quitting behaviour. In the following sections, we first present the development of models using DT, RS, and NN techniques. We then compare the performance of these models in terms of their predictive abilities. With the models built, we identify the significant attributes that contribute to the smokers' quitting status. We then discuss how the results of this study can be used as guidelines for making effective tobacco control policies.

2 Building Prediction Models

2.1 The International Tobacco Control (ITC) Data

The data used in this study is from the first year of ITC-4C survey and it consists of 5,450 smokers. Among the smokers, 37.22% made a quit attempt and 75.89% of them planned to quit. This implies that some smokers who planned to quit did not make a quit attempt within a certain period of time. Prior to actually quitting, significant steps taken by most smokers include planning to quit (PTQ) and making a quit attempt (MQA). In this paper, we refer to these 2 possible states as being the "status" of the smokers. From the ITC-4C data set we know more specifically 90.5% people who made a quit attempt actually planned quit and 89.3% people who ultimately quit actually planned to quit.

We use MQA or PTQ as the output attribute for our models. As input attributes, we use 13 smokers' beliefs about quitting (BQ) attributes and 13 smokers' beliefs about smoking (BS) attributes. Table 1 summarizes the input and output attributes.

Table 1. Input and output attributes of prediction models

Input attributes			
Beliefs about quitting (BQ)		Beliefs about smoking (BS)	
(BQ01)	Concern for personal health	(BS01)	Enjoyment of smoking
(BQ02)	Smoking effecting non-smokers	(BS02)	Regret of smoking
(BQ03)	Society disapproves of smoking	(BS03)	Smoking reduces stress
(BQ04)	Cigarette price	(BS04)	Too much money on cigarettes
(BQ05)	Smoking restrictions at work	(BS05)	Smoking is important part of life
(BQ06)	Smoking restrictions in public places	(BS06)	Smoking helps control weight
(BQ07)	Advice from doctor	(BS07)	Mixed emotions about smoking
(BQ08)	Stop-smoking medication	(BS08)	Impact on people important to you
(BQ09)	Quit line advice	(BS09)	Availability of comfortable smoking places
(BQ10)	Advertisement about quitting	(BS10)	Society disapproves of smoking
(BQ11)	Warning labels on cigarette packages	(BS11)	Medical evidence exaggerates\ smoking harm
(BQ12)	Example for children	(BS12)	Enjoyment considering the inevitability of death
(BQ13)	Benefits from quitting	(BS13)	Smoking is no more risky than other risky activities
Output attribute			
Making a quit attempt (MQA)		Planning to quit (PTQ)	

2.2 Techniques

Decision trees (DT) are a rule-based recursive structure to perform sequential classification and investigate suitable attributes for prediction. Decision trees can handle a large number of data quickly and allow missing values to be included without pre-processing [6]. Rough sets (RS) are an intelligent mathematical tool to deal with vague and uncertain data [7]. The rough sets are based on the premise that lowering the degree of precision in the data makes the data pattern more visible [8], whereas the central premise of the rough set philosophy is that the knowledge consists in the ability of classification. Neural networks are widely applied to model non-linear data and examine the relationship between input and output attributes [9]. Neural networks (NN) consider all input attributes and can be applied to complex modeling. In this paper, we vary input attribute categories to investigate the prediction accuracy and important attributes of DT, RS and NN models.

2.3 Models

We conduct a set of experiments to identify the internal relationships between the two quit statuses (MQA and PTQ) of the smokers.

For the first set of experiments, we use PTQ as our output attribute. For each technique, the input attributes are split into three groups: 13 BQ attributes, 13 BS attributes, and 13 BQ attributes plus 13 BS attributes.

In the second stage, we use MQA as the output attribute for prediction using the above three groups of attributes as inputs. In total, we build 18 models for comparison and analysis.

3 Performance Evaluation

Table 2 shows the prediction rates of DT, RS and NN models generated by the 6 different data sets. All NN models show the highest prediction rates for all data sets, whereas RS models have the lowest rates.

For input attributes of three models, combining BQ and BS attributes together improves the prediction accuracy slightly than using them separately as inputs for all three models. As we can see from three models, for predicting MQA, the models developed using BS attributes has better prediction rates than the ones using BQ

Table 2. Prediction rates of 18 models

Model	Output attributes	Input attributes		
		BQ	BS	BQ + BS
Decision tree	MQA	62.60%	64.00%	64.90%
	PTQ	80.60%	75.40%	81.30%
Rough set	MQA	56.78%	58.94%	60.53%
	PTQ	75.56%	71.74%	78.36%
Neural network	MQA	64.51%	64.94%	66.89%
	PTQ	82.26%	79.82%	82.83%

attributes. For predicting PTQ, the accuracy rate of the models using BQ attributes is better than the ones using BS attributes. This indicates that the set of BS attributes is a better prediction for MQA whilst BQ is a better prediction for PTQ.

In all models, each model has a better predictive accuracy on PTQ than MQA. The neural network model developed using both BQ and PS attributes for predicting PTQ has the highest predictive ability at 82.83%.

4 Discussion

To develop a better prediction model for predicting the effects of tobacco control policies, we conducted a performance evaluation study of intelligent models. 18 prediction models including decision trees, rough sets, and neural networks were developed using different sets of BQ and BS attributes as inputs for two output variables, MQA and PTQ. The accuracy rates of three models were compared with in terms of their predictive ability. The results of the experiments have practical and theoretical implications for tobacco control policy implementing intelligent techniques. The findings of the study are summarized as follows.

4.1 Selection of Prediction Models

Concerning choice of intelligent techniques, the findings of the study will help tobacco policy decision makers choose the better prediction model for given tobacco control policies. As shown in Table 2, the accuracy rates of NN models are better than that of DT and RS models for all cases.

This, therefore, suggests that neural networks are more suitable than the other two techniques for predicting smokers' quitting behaviours by identifying the key determinants of them.

4.2 Determinate Data Sets for Predicting PTQ or MQA

The experimental result shows that the accuracy rates of the models using both BQ and BS attributes as inputs are better than the models using only BQ or BS attributes for predicting the PTQ or MQA status of smokers. This suggests that tobacco control policy decision makers who aim at developing effective tobacco control policies need to consider addressing issues that have impacts on both BQ ("beliefs about quitting") and BS ("belief about smoking") attributes of smokers.

The models using BS attributes for predicting MQA has better prediction rates than the ones using BQ attributes, while the accuracy rates of the models using BQ attributes for predicting PTQ are better than the ones using BS attributes. This indicates that smokers' beliefs about smoking contribute more to smokers who make a quit attempt, whereas smokers' beliefs about quitting have more impact on smokers who plan to quit. Furthermore, three different cases of input attributes for the output attribute PTQ have higher accuracy rates than the ones for MQA. This is not surprising since the proportion of smokers who is planning to quit (PTQ) is much higher than those making a quit attempt (MQA).

The above findings about determinant input data sets for predicting MQA and PTQ suggests that tobacco control policies made with reference to these determinant input data sets can considerably influence smokers' quitting status as identified in this

study. The use of various input data sets can be further investigated in order to develop better prediction models as a future research.

4.3 Influential Input Attributes for Predicting PTQ or MQA

Table 3 shows the influential input attributes for predicting MQA and PTQ respectively from 18 models, based on different data sets. Input attributes for each data set are ranked based on their contribution to prediction ability of each model. The highlighted input attributes indicate the common influential attributes from the DT, RS and NN models for each same data set.

The way each technique identifies the important attributes differs. For DT models, attribute usage determines the most significant attributes. For RS models, classification quality gives as a measure of attribute significance. For NN models, attribute weight gives as a measure of attribute significance. Although the techniques use a different approach, it is interesting to note that there are some common attributes which all 3 techniques agreed as being important.

Significant BQ attributes. If we restrict the input data set to BQ attributes, all techniques agree that for predicting MQA and PTQ the following attributes are in the top 5 significant attributes, BQ13, BQ01, and BQ04. Also BQ12 is in the most significant 5 apart from DT models for MQA. This suggests that the most significant factors influencing a smoker to plan to quit and to make a quit attempt are their knowledge of the benefits of quitting, their concern for their own personal health, their concern regarding the example they are setting for children, and the cost of cigarettes. Thus a tobacco control policy which is aimed at educating smokers about the impact smoking has on their own and their children's long term health and quality of life combined with cigarette prices that add a significant incentive to quit, appear to be the best basis for encouraging smokers to plan to quit and to make a quit attempt.

Significant BS attributes. The top 6 BS attributes for predicting MQA include two common attributes, BS01 and BS07, whereas the significant BS attributes for predicting PTQ show two common attributes, BS01 and BS12. For a smoker to plan to quit, enjoyment of smoking and feelings about the inevitability of death are significant factors. Both of these are either very difficult or impossible to influence via policy. For a smoker to make a quit attempt, enjoyment of smoking and mixed emotions about smoking are significant factors. This suggests that if a smoker can be given an incentive to reconsider their smoking behaviour, they may be induced to make a quit attempt.

Significant BQ+BS attributes. When we combine the BQ and BS attributes as inputs for predicting MQA, both DT and RS models agree that BS01, BQ04, BS13, and BS11 are in the top 9 significant factors for a smoker to make a quit attempt and plan to quit. In addition, when we combine the BQ and BS attributes as inputs for predicting PTQ, both DT and RS models agree that BS01, BQ04, and BS13 are in the top 9 significant factors for a smoker to make a quit attempt and plan to quit. Influencing a smoker's enjoyment of smoking is difficult to accomplish. However, if we consider the other 3 significant factors, we can suggest that tobacco control policies aimed at educating smokers of the harm involved with smoking and convincing them that evidence of the harm resulting from smoking has not been exaggerated, combined with judicious cigarette pricing, is likely to encourage a smoker to make a quit attempt and/or plan to quit.

Table 3. Influential input attributes of DT, RS and NN models

Output attribute: MQA					
Input attribute: BQ					
Decision Tree		Rough Sets		Neural Networks	
100%	BQ01	0.124	BQ13	0.10355	BQ01
53%	BQ11	0.086	BQ04	0.08508	BQ04
32%	BQ02	0.079	BQ01	0.08275	BQ08
27%	BQ13	0.078	BQ02	0.08213	BQ12
26%	BQ04	0.076	BQ12	0.08192	BQ13
Input attribute: BS					
Decision Trees		Rough Sets		Neural Networks	
72%	BS01	0.056	BS01	0.10604	BS08
70%	BS07	0.047	BS06	0.08817	BS01
51%	BS02	0.038	BS12	0.08643	BS02
45%	BS13	0.038	BS13	0.08497	BS04
10%	BS08	0.036	BS05	0.07949	BS07
		0.036	BS07	0.07585	BS03
Input attribute : BQ + BS					
Decision Trees		Rough Sets		Neural Networks	
100%	BQ01	0.134	BQ13	0.04619	BS08
60%	BS01	0.119	BS01	0.04615	BQ01
39%	BQ06	0.108	BS06	0.04490	BS02
39%	BQ12	0.098	BS05	0.04392	BQ13
28%	BS13	0.096	BQ04	0.04342	BS03
26%	BS08	0.091	BQ10	0.04263	BS01
22%	BS10	0.091	BS13	0.04207	BS05
2%	BS11	0.082	BS11	0.04052	BQ09
1%	BQ04	0.071	BS02	0.04035	BS07
Output attribute: PTQ					
Input attribute: BQ					
Decision Trees		Rough Sets		Neural Networks	
68%	BQ13	0.109	BQ13	0.11354	BQ01
62%	BQ01	0.065	BQ04	0.08179	BQ04
50%	BQ04	0.064	BQ01	0.08074	BQ12
50%	BQ10	0.063	BQ12	0.07956	BQ13
49%	BQ12	0.051	BQ02	0.07897	BQ02
Input attribute: BS					
Decision Trees		Rough Sets		Neural Networks	
93%	BS08	0.045	BS01	0.10264	BS01
69%	BS12	0.041	BS06	0.09481	BS12
63%	BS01	0.033	BS13	0.08890	BS08
23%	BS04	0.032	BS05	0.08377	BS04
6%	BS06	0.030	BS12	0.08313	BS02
6%	BS11	0.028	BS11	0.07501	BS05
Input attribute: BQ + BS					
Decision Trees		Rough Sets		Neural Networks	
65%	BQ10	0.062	BQ13	0.04929	BQ01
62%	BQ01	0.033	BQ12	0.04747	BQ13
52%	BS01	0.032	BQ04	0.04530	BS01
51%	BS13	0.029	BS01	0.04283	BS04
43%	BQ11	0.026	BQ08	0.04254	BQ12
42%	BQ04	0.024	BS06	0.04193	BS07
19%	BQ03	0.020	BS13	0.04012	BQ06

5 Conclusion

In this paper, we have built 18 models using decision trees, rough sets and neural networks, using 6 data sets created from the ITC-4C data. Smokers' beliefs about quitting and beliefs about smoking are used as input attributes for predicting their status of making a quit attempt or planning to quit. The models developed provide useful information about the most significant factors encouraging a smoker to plan to quit and/or make a quit attempt. The common factors identified from these models indicate the knowledge a smoker has regarding the damage caused by smoking, the impact on the smokers' health and well being as well as that of their children and cigarette price. Also the credibility of the medical evidence regarding the information provided to smokers is important. We have discussed the possible use of these results for tobacco control policy makers who wish to develop effective policies that will have significant impacts on smokers' decisions as to whether to try and quit smoking. The outcomes of this study provide new insights into how to best address the challenging issue of predicting the effects of tobacco control polices.

References

1. Thompson, M.E., Fong, G.T., Hammond, D., Boudreau, C., Driezen, P., Hyland, P., Borland, R., Cummings, K.M., Hastings, G.B., Siahpush, M., Machintosh, A.M., Laux, F.L.: Methods of the International Tobacco Control (ITC) Four Country Survey. *Tobacco Control* 15(3), iii12–iii18 (2006)
2. Ding, X., Bedingfield, S., Yeh, C.-H., Zhang, J., Petrovic-Lazarevic, S., Coghill, K., Borland, R., Young, D.: A Decision Tree Approach for Predicting Smokers' Quit Intentions. In: *The 2008 International Conference on Communications, Circuits and Systems*, pp. 1159–1163 (2008)
3. Hammond, D., Fong, G.T., McNeil, A., Borland, R., Cummings, K.M.: Effectiveness of Cigarette Warning Labels in Informing Smokers about the Risks of Smoking: Findings from the International Tobacco Control (ICT) Four Country Survey. *Tobacco Control* 15(3), iii19–iii25 (2006)
4. Harris, F., MacKintosh, A.M., Anderson, S., Hastings, G., Borland, R., Fong, G.T., Hammond, D., Cummings, K.M.: Effects of the 2003 Advertising/Promotion Ban in the United Kingdom on Awareness of Tobacco Marketing: Findings from the International Tobacco Control (ITC) Four Country Survey. *Tobacco Control* 15(3), iii26–iii33 (2006)
5. Hyland, A., Borland, R., Li, Q., Yong, H.-H., McNeill, A., Fong, G.T., O'Connor, R.J., Cummings, K.M.: Individual-Level Predictors of Cessation Behaviours among Participants in the International Tobacco Control (ITC) Four Country Survey. *Tobacco Control* 15(3), iii83–iii94 (2006)
6. Quinlan, J.R.: Generating Production Rules from Decision Trees, <http://dli.iit.ac.in/ijcai/IJCAI-87-VOL1/PDF/063.pdf>
7. Pawlak, Z., Grzymala-Busse, J., Slowinski, R., Ziarko, W.: Rough Sets. *Communications of the ACM* 38(11), 89–95 (1995)
8. Slowinski, R.: Intelligent Decision Support. In: *Handbook of Applications and Advances of the Rough Sets Theory*, Kluwer Academic Publishers, Dordrecht (1992)
9. Yeh, C.-H., Lin, Y.-C.: Neural Network Models for Transforming Consumer Perception into Product Form Design. In: Wang, J., Yi, Z., Żurada, J.M., Lu, B.-L., Yin, H. (eds.) *ISSN 2006. LNCS*, vol. 3973, pp. 799–804. Springer, Heidelberg (2006)

Range Facial Recognition with the Aid of Eigenface and Morphological Neural Networks

Chang-Wook Han

Department of Electrical Engineering, Dong-Eui University,
995 Eomgwangno, Busanjin-gu, Busan, 614-714, South Korea
cwhan@deu.ac.kr

Abstract. The depth information in the face represents personal features in detail. In particular, the surface curvatures extracted from the face contain the most important personal facial information. These surface curvature and eigenface, which reduce the data dimensions with less degradation of original information, are collaborated into the proposed 3D face recognition algorithm. The principal components represent the local facial characteristics without loss for the information. Recognition for the eigenface referred from the maximum and minimum curvatures is performed. To classify the faces, the max plus algebra based neural networks (morphological neural networks) optimized by hybrid genetic algorithm are considered. Experimental results on a 46 person data set of 3D images demonstrate the effectiveness of the proposed method.

1 Introduction

Today's computer environments are changing because of the development of intelligent interface and multimedia. To recognize the user automatically, people have researched various recognition methods using biometric information – fingerprint, face, iris, voice, vein, etc [1]. In a biometric identification system, the face recognition is a challenging area of research, next to fingerprinting, because it is a no-touch style. For visible spectrum imaging, there have been many studies reported in the literature [2]. But the method has been found to be limited in their application. It is influenced by lighting illuminance and encounters difficulties when the face is angled away from the camera. These factors cause low recognition. To solve these problems a computer company has developed a 3D face recognition system [2][3]. To obtain a 3D face, this method uses stereo matching, laser scanner, etc. Stereo matching extracts 3D information from the disparity of 2 pictures which are taken by 2 cameras. Even though it can extract 3D information from near and far away, it has many difficulties in practical use because of its low precision. 3D laser scanners extract more accurate depth information about the face, and because it uses a filter and a laser, it has an advantage of not being influence by the lighting illuminance when it is angled away from the camera. A laser scanner can measure the distance, therefore, a 3D face image can be reduced by a scaling effect that is caused by the distance between the face and the camera [4][5]. Thus the use of 3D face image is now being more readily researched [3][6-10].

One of the most successful techniques of face recognition as statistical method is principal component analysis (PCA), and specifically eigenfaces [11][12]. In this paper,

we introduce a novel face recognition method for eigenfaces using the curvature that well presenting personal characteristics and reducing dimensional spaces.

Neural networks have been successfully applied to face recognition problems [13]. In this paper, the morphological neural networks (MNNs) based on max plus algebra [14][15] are considered to identify the faces. However, the complexity of the MNNs increases exponentially with the parameter values, i.e. input number, output number, hidden neuron number, etc., and becomes unmanageable. To optimize these complex MNNs, the memetic algorithm (hybrid genetic algorithm) [16] is also considered rather than the gradient-based learning methods because of its poor convergence properties.

2 Surface Curvature

For each data point on the facial surface, the principal, Gaussian and mean curvatures are calculated and the signs of those (positive, negative and zero) are used to determine the surface type at every point. The $z(x, y)$ image represents a surface where the individual Z -values are surface depth information. Here, x and y is the two spatial coordinates. We now closely follow the formalism introduced by Peet and Sahota [17], and specify any point on the surface by its position vector:

$$R(x, y) = xi + yj + z(x, y)k \tag{1}$$

The first fundamental form of the surface is the expression for the element of arc length of curves on the surface which pass through the point under consideration. It is given by:

$$I = ds^2 = dR \cdot dR = Edx^2 + 2Fdxdy + Gdy^2 \tag{2}$$

where

$$E = 1 + \left(\frac{\partial z}{\partial x}\right)^2, \quad F = \frac{\partial z}{\partial x} \frac{\partial z}{\partial y}, \quad G = 1 + \left(\frac{\partial z}{\partial y}\right)^2 \tag{3}$$

The second fundamental form arises from the curvature of these curves at the point of interest and in the given direction:

$$II = edx^2 + 2fdxdy + gdy^2 \tag{4}$$

where

$$e = \frac{\partial^2 z}{\partial x^2} \Delta, \quad f = \frac{\partial^2 z}{\partial x \partial y} \Delta, \quad g = \frac{\partial^2 z}{\partial y^2} \Delta \tag{5}$$

and

$$\Delta = (EG - F^2)^{-1/2} \tag{6}$$

Casting the above expression into matrix form with;

$$V = \begin{pmatrix} dx \\ dy \end{pmatrix}, \quad A = \begin{pmatrix} E & F \\ F & G \end{pmatrix}, \quad B = \begin{pmatrix} e & f \\ f & g \end{pmatrix} \tag{7}$$

the two fundamental forms become:

$$I = V'AV \quad I = V'BV \tag{8}$$

Then the curvature of the surface in the direction defined by V is given by:

$$k = \frac{V'BV}{V'AV} \tag{9}$$

Extreme values of k are given by the solution to the eigenvalue problem:

$$(B - kA)V = 0 \tag{10}$$

or

$$\begin{vmatrix} e - kE & f - kF \\ f - kF & g - kG \end{vmatrix} = 0 \tag{11}$$

which gives the following expressions for k_1 and k_2 , the minimum and maximum curvatures, respectively:

$$k_1 = \left\{ gE - 2Ff + Ge - [(gE + Ge - 2Ff)^2 - 4(eg - f^2)(EG - F^2)]^{1/2} \right\} / 2(EG - F^2) \tag{12}$$

$$k_2 = \left\{ gE - 2Ff + Ge + [(gE + Ge - 2Ff)^2 - 4(eg - f^2)(EG - F^2)]^{1/2} \right\} / 2(EG - F^2) \tag{13}$$

Here we have ignored the directional information related to k_1 and k_2 , and chosen k_2 to be the larger of the two. For the present work, however, this has not been done. The two quantities, k_1 and k_2 , are invariant under rigid motions of the surface. This is a desirable property for us since the cell nuclei have no predefined orientation on the slide (the $x - y$ plane).

The Gaussian or total curvature K is defined by

$$K = k_1k_2 \tag{14}$$

and the mean curvature M is defined by

$$M = (k_1 + k_2) / 2 \tag{15}$$

which gives k_1 and k_2 , the minimum and maximum curvatures, respectively. It turns out that the principal curvatures, k_1 and k_2 , and Gaussian are best suited to the detailed characterization for the facial surface. For the simple facet model of second order polynomial of the form, i.e. an 3x3 window implementation in our range images, the local region around the surface is approximated by a quadric

$$z(x, y) = a_{00} + a_{10}x + a_{01}y + a_{01}y + a_{20}x^2 + a_{02}y^2 + a_{11}xy \tag{16}$$

and the practical calculation of principal and Gaussian curvatures is extremely simple.

3 Eigenface

3.1 Computing Eigenfaces

Consider face images of size $N \times N$, extracted contour line value. These images can be thought as a vector of dimension N^2 , or a point in N^2 – dimensional space. A set of images, therefore, corresponds to a set of points in this high dimensional space. Since facial images are similar in structure, these points will not be randomly distributed, and therefore can be described by a lower dimensional subspace. Principal component analysis gives the basis vectors for this subspace. Each basis vector is of length N^2 , and is the eigenvector of covariance matrix corresponding to the original face images. Let $\Gamma_1, \Gamma_2, \dots, \Gamma_M$ be the training set of face images. The average face is defined by

$$\Psi = \frac{1}{M} \sum_{n=1}^M \Gamma_n \quad (17)$$

Each face differs from the average face by the vector $\Phi_i = \Gamma_n - \Psi$. The covariance matrix

$$C = \frac{1}{M} \sum_{n=1}^M \Phi_n \Phi_n^T \quad (18)$$

has a dimension of $N^2 \times N^2$. Determining the eigenvectors of C for typical size of N is intractable task. Once the eigenfaces are created, identification becomes a pattern recognition task. Fortunately, we determine the eigenvectors by solving an $M \times M$ matrix instead.

3.2 Identification

The eigenfaces span an M -dimensional subspace of the original N^2 image space. The M significant eigenvectors are chosen as those with the largest corresponding eigenvalues. A test face image Γ is projected into face space by the following operation: $\omega_n = u_n^T (\Gamma - \Psi)$, for $n=1, \dots, M$, where u_n is the eigenvectors for C . The weights ω_n from a vector $\Omega^T = [\omega_1 \ \omega_2 \ \dots \ \omega_M]$ which describes the contribution of each eigenface in representing the input face image. This vector can then be used to fit the test image to a predefined face class. A simple technique is to use the Euclidian distance $\varepsilon_n = \|\Omega - \Omega_n\|$, where Ω_n describes the n th face class. In this paper, we consider the morphological neural networks to compare with the distance as described next section.

4 Morphological Neural Networks and Its Optimization

Morphological neural networks (MNNs) are constructed based on morphological operators [14][15] which are defined as, ‘max{ a, b}’ and ‘a+b’, instead of standard addition and multiplication in ordinal algebra, respectively. This paper considers the MNNs with three layers (input, middle, and output layers), as shown in Fig. 1, where the input, middle, and output layer vectors with N_I, N_m , and N_O dimensions, respectively.

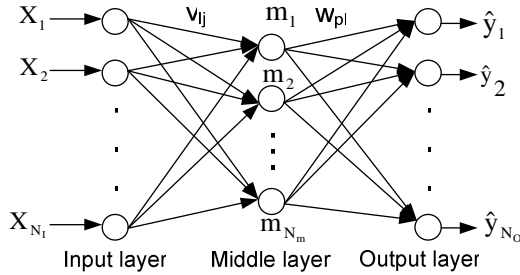


Fig. 1. Topology of the morphological neural networks

According to the definition of morphological operations, the middle layer vector can be calculated as

$$m_l = \max_{j=1}^{N_i} \{v_{lj} + x_j\}, \quad l = 1, 2, \dots, N_m \quad (19)$$

and the output layer vector is calculated as

$$\hat{y}_p = \max_{l=1}^{N_m} \{w_{pl} + m_l\}, \quad p = 1, 2, \dots, N_o \quad (20)$$

To optimize the connection weights of the MNNs, the memetic algorithm [16] is also considered rather than the gradient-based learning methods because of its poor convergence properties. As proved in [16], the memetic algorithms are more effective than the optimization scenario of other genetic algorithms. Therefore, the optimization scenario in [16] will be considered in this approach.

5 Experimental Results

In this study, we used a 3D laser scanner made by a 4D culture to obtain a 3D face image. First, a laser line beam was used to strip the face for 3 seconds, thereby obtaining a laser profile image, that is, 180 pieces and no glasses. The obtained image size was extracted by using the extraction algorithm of line of center, which is 640 x 480. Next, calibration was performed in order to process the height value, resampling and interpolation. Finally, the 3D face images for this experiment were extracted, at 320x320. A database is used to compare the different strategies and is composed of 92 images (two images of 46 persons). Of the two pictures available, the second photos were taken at a time interval of 30 minutes.

From these 3D face images, we found the nose tip point and contour line threshold values (for which the fiducial point is nose tip), and subsequently extracted images around the nose area. To perform recognition experiments for extracted area, we first need to create two sets of images, i.e. training and testing. For each of the two views, 46 normal-expression images were used as the training set. Training images were used to generate an orthogonal basis into which each 3D image in training data set is projected. Testing images are a set of 3D images extracted local area we wish to identify.

Table 1. The comparison of the recognition rate (%)

		Best1	Best5	Best10	Best15
k ₁	MNNs	54.7	67.8	79.6	86.8
	k-NN	42.9	57.1	66.7	66.7
k ₂	MNNs	64.0	81.5	85.9	91.1
	k-NN	61.9	78.5	83.3	88.1

Once the data sets have been extracted with the aid of eigenface, the development procedure of the MNNs should be followed for the face recognition. The considered parameter values for the MNNs are describes as follows: number of input node=46, number of hidden node=12, number of output node=46, range of connection weights=[-0.5, 0.5], and the data sets are normalized in [0, 1]. The used parameter values for the optimization are the same as [16] except the population size (500) and maximum generation number (1000). To apply the MNNs to classification problems, the output (class) should be discretized as binary. For example, if we assume that there are 5 classes (5 persons) in the data sets, the number of output crisp set should be 5. If the person belongs to the 2nd-class, the Boolean output can be discretized as “0 1 0 0 0”. In this classification problem, the winner-take-all method is used to decide the class of the testing data set. This means that the testing data are classified as the class which has the biggest output value. To evaluate the fitness, the root mean square error (RMSE) was used. Since a genetic algorithm is a stochastic optimization method, ten times independent simulations were performed to compare the results with the conventional classification methods, as described in Table 1 and Fig. 2. In Table 1 and Fig. 2, the results of the MNNs are averaged over ten times independent simulations, and subsequently compared with the results of the conventional method (k-nearest neighborhood: k-NN).

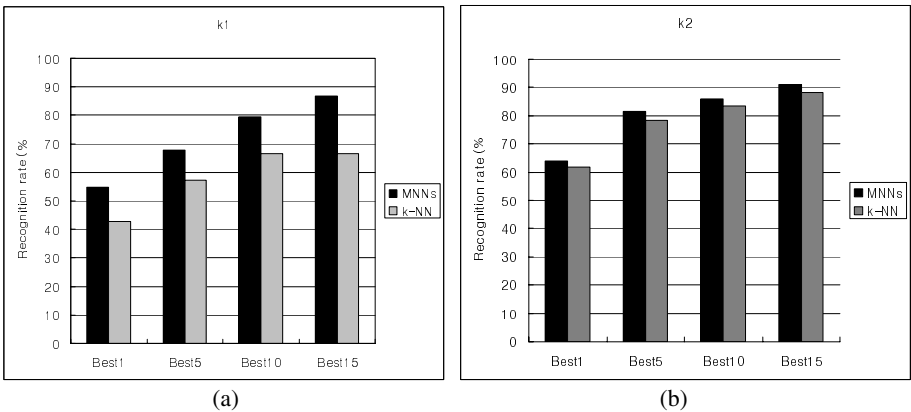


Fig. 2. The recognition results using eigenfaces for each areas: (a) k₁, (b) k₂

6 Conclusions

The surface curvatures extracted from the face contain the most important personal facial information. We have introduced, in this paper, a new practical implementation of a person verification system using the local shape of 3D face images based on eigenfaces and MNNs. The underlying motivations for our approach originate from the observation that the curvature of face has different characteristic for each person. We found the exact nose tip point by using an iterative selection method. The low-dimensional eigenfaces represented were robust for the local area of the face. To classify the faces, the MNNs were used. Experimental results on a group of face images (92 images) demonstrated that our approach produces excellent recognition results for the local eigenfaces.

From the experimental results, we proved that the process of face recognition may use low dimension, less parameters, calculations and less same person images (used only two) than earlier suggested. We consider that there are many future experiments that could be done to extend this study.

References

1. Jain, L.C., Halici, U., Hayashi, I., Lee, S.B.: Intelligent biometric techniques in fingerprint and face recognition. CRC Press, Boca Raton (1999)
2. 4D Culture, <http://www.4dculture.com>
3. Cyberware, <http://www.cyberware.com>
4. Chellapa, R., et al.: Human and Machine Recognition of Faces: A Survey. UMCP CS-TR-3399 (1994)
5. Hallinan, P.L., Gordon, G.G., Yuille, A.L., Giblin, P., Mumford, D.: Two and three dimensional pattern of the face. A K Peters Ltd (1999)
6. Chua, C.S., Han, F., Ho, Y.K.: 3D Human Face Recognition Using Point Signature. In: Proc. of the 4th ICAFG (2000)
7. Tanaka, H.T., Ikeda, M., Chiaki, H.: Curvature-based face surface recognition using spherical correlation. In: Proc. of the 3rd IEEE Int. Conf. on Automatic Face and Gesture Recognition, pp. 372–377 (1998)
8. Gordon, G.G.: Face Recognition based on depth and curvature feature. In: Proc. of the IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, pp. 808–810 (1992)
9. Chellapa, R., Wilson, C.L., Sirohey, S.: Human and machine recognition of faces: A survey. Proceedings of the IEEE 83(5), 705–740 (1995)
10. Lee, J.C., Milios, E.: Matching range image of human faces. In: Proc. of the 3rd Int. Conf. on Computer Vision, pp. 722–726 (1990)
11. Turk, M., Pentland, A.: Eigenfaces for Recognition. Journal of Cognitive Neuroscience 3(1), 71–86 (1991)
12. Heshner, C., Srivastava, A., Erlebacher, G.: Principal Component Analysis of Range Images for Facial Recognition. In: Proc. of CISST (2002)
13. Zhao, Z.Q., Huang, D.S., Sun, B.Y.: Human face recognition based on multi-features using neural networks committee. Pattern Recognition Letters 25, 1351–1358 (2004)
14. Davidson, J.L., Ritter, G.X.: A Theory of Morphological Neural Networks. SPIE 1215, 378–388 (1990)

15. Ritter, G.X., Li, D., Wilson, J.N.: Image Algebra and its Relationship to Neural Networks. SPIE 1098, 90–101 (1989)
16. Han, C.W., Park, J.I.: SA-selection-based Genetic Algorithm for the Design of Fuzzy Controller. *International Journal of Control, Automation, and Systems* 3(2), 236–243 (2005)
17. Peet, F.G., Sahota, T.S.: Surface Curvature as a Measure of Image Texture. *IEEE Trans. PAMI* 7(6), 734–738 (1985)

Modular Bayesian Network Learning for Mobile Life Understanding

Keum-Sung Hwang and Sung-Bae Cho

Department of Computer Science, Yonsei University, Seoul, Korea
{yellowg, sbcho}@cs.yonsei.ac.kr

Abstract. Mobile devices can now handle a great deal of information thanks to the convergence of diverse functionalities. Mobile environments have already shown great potential in terms of providing customized services to users because they can record meaningful and private information continually for long periods of time. Until now, most of this information has been generally ignored because of the limitations of mobile devices in terms of power, memory capacity and speed. In this paper, we propose a novel method that efficiently infers semantic information and overcome the problems. This method uses an effective probabilistic Bayesian network model for analyzing various kinds of log data in mobile environments, which were modularized in this paper to decrease complexity. We also discuss how to discover and update the Bayesian inference model by using the proposed BN learning method with training data. The proposed methods were evaluated with artificial mobile log data generated and collected in the real world.

Keywords: Modularized Probabilistic Reasoning, Mobile Application.

1 Introduction

Mobile environments have very different characteristics from desktop computer environments. First of all, mobile devices can collect and manage various kinds of user information, for example, by logging a user's calls, SMS (short message service), photography, music-playing and GPS (global positioning system) information. Also, mobile devices can be customized to fit any given user's preferences. Furthermore, mobile devices can collect everyday information effectively. Such features allow for the possibility of diverse and convenient services, and have attracted the attention of researchers and developers. Recent research conducted by Nokia is a good example [1]. Especially, the context-aware technique that has recently been widely researched is more applicable to mobile environments, so many intelligent services such as intelligent calling services [2], messaging services [3], analysis, collection and management of mobile logs [4-6] have been actively investigated.

However, mobile devices do present some limitations. They contain relatively insufficient memory capacity, lower CPU power (data-processing speed), smaller screen sizes, awkward input interfaces, and limited battery lives when compared to

desktop PCs. In addition, they have to operate in the changeable real world, which means that they require more active and effective adaptation functions [7].

In this paper, we propose a novel way of analyzing mobile log data effectively and extracting semantic information. The proposed method adopts a Bayesian probabilistic model to efficiently manage various uncertainties that can occur when working with mobile environments. The proposed method uses a cooperative reasoning method with modular Bayesian network (BN) model in order to work competently in mobile environments and contains how to discover and update the modular BNs from training data based on the previous work [8].

There have already been various attempts to analyze log data and to support expanded services by using the probabilistic approach. Krause, *et al.* collected on mobile devices and estimated the user's situation in order to provide smart services [9]. Horvitz, *et al.* proposed a method that detected and estimated landmarks by discovering a given human's cognitive activity model from PC log data based on the Bayesian approach [10]. However, these methods were not suitable for mobile devices that were limited in terms of capacity and power. For larger domains, the general BN model requires highly complex computation. This is a crucial problem when it comes to modeling everyday life situations with mobile devices.

To overcome these problems, a more appropriate approach was necessary. The following researchers have studied methods of reducing the levels of complexity. Marenconi, *et al.* [11] tried to reduce the complexity levels of the BN model by dividing it into several multi-level modules and using procedural reasoning of the connected BNs (just like chain inference). However, this method required procedural and classified properties of the target functions.

Tu, *et al.* [12] proposed a hybrid BN model that allowed hierarchical hybridization of BNs and HMMs. However, it supported only links from lower level HMMs to higher level BNs without consideration of links between same level BNs. They also remained the hybridization of low and high level BNs as future works.

2 Landmark Reasoning from Mobile Log Data

The overall process of landmark reasoning from the mobile log data used in this paper is shown in Fig. 1. Various mobile log data is preprocessed in advance, and then the landmark-reasoning module detects the landmarks. The preprocessing module is operated by the techniques of pattern recognition and simple rule reasoning. The BN reasoning module performs probabilistic inference.

We used modular Bayesian network model proposed in the prior work [8] since it can manage the modularized Bayesian networks. It considers the co-causality of the modularized BN by n -pass cooperative reasoning [8] with a virtual linking technique, which is performed to add the virtual nodes and regulate their conditional probability values (CPVs) to apply the probability of the evidence.

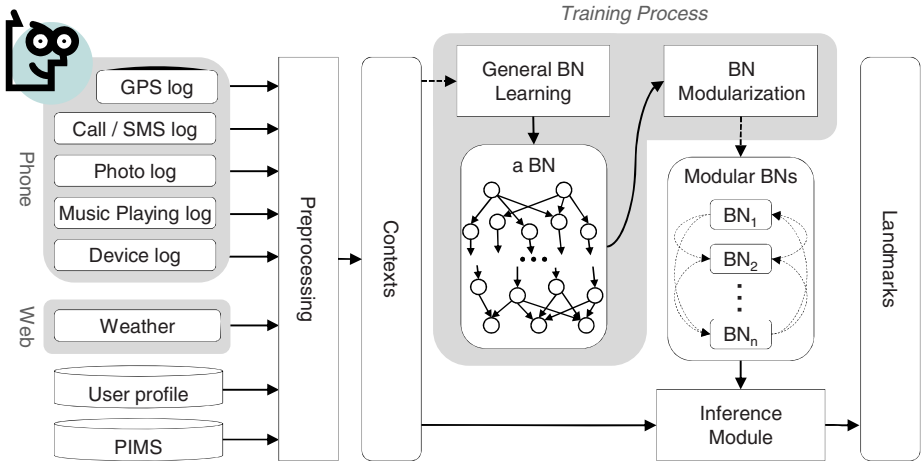


Fig. 1. The process of the landmark extraction from mobile log data

3 Modular Bayesian Networks Modeling

The whole procedure for the modular BN learning method is shown in Fig. 2. The method includes a structure learning, modularization, and parameter learning processes.

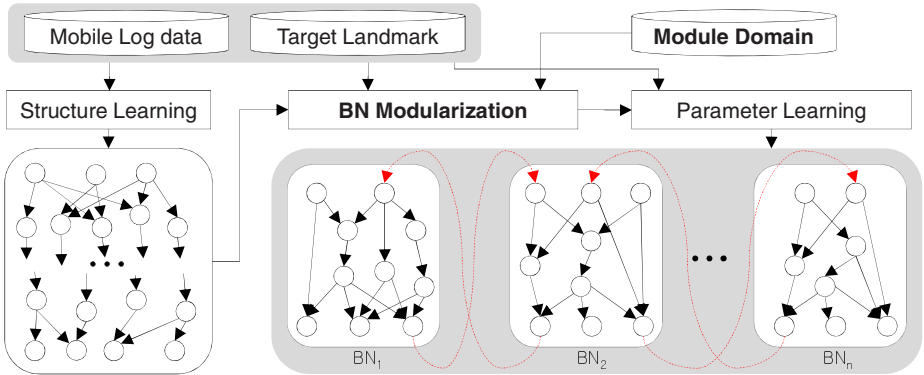


Fig. 2. The learning procedure of modular Bayesian networks

3.1 Bayesian Network Structure Learning Method

We adopted the domain-constrained BN Structure Learning method [13] for BN structure learning, which aims to discover the structure of the BN hierarchically with a constrained parent domain of nodes. The algorithm limits the hierarchical level at which a node can be positioned and reduces the search domain for the parent of the nodes. It can control the general direction of causality and decrease the searching complexity. In this paper, we use the algorithm to exclude the arcs between the

Table 1. The level and domain definition

Level	Node Domain	Parent Domain
0	$LM=\{lm_1,lm_2,\dots\}$	$D_0=\Phi$
1	$L=\{l_1,l_2,\dots\}$	$D_1=L \cup LM$

LM is landmark set, L is log context set.

evidence nodes (but we permits arcs between the landmarks.) and maintain the direction of the arcs in order to use the virtual link technique. Table 1 shows the level and domain settings used in this paper.

Fig. 3 shows the proposed domain-constrained K2 algorithm, based on the K2 algorithm as proposed by Cooper and Herskovits [14], which is the most popular BN algorithm and the basis of many advanced discovery algorithms. This algorithm adopts a score metric known as the K2 metric, which calculates scores based on the difference between the BN graph and the training data distribution. The search process of the K2 algorithm is greedy and heuristic.

```

Sort_nodes_by_topological_order(X, O)
for i=1 to n do:
     $\pi_i = \Phi$  // initialize the parent set
    Score[i]=g(xi,  $\pi_i$ ) // k2 metric score of node xi
    Continue=true
    repeat
        Z= arg maxj g(xi,  $\pi_i \cup \{x_j\}$ ) and j<i and xj ∉  $\pi_i$  and xj ∈ Dlevel(i)
        Score'[i]= g(xi,  $\pi_i \cup \{x_z\}$ )
        if Score'[i]>Score[i] then
            Score[i] = Score'[i],  $\pi_i = \pi_i \cup \{x_z\}$ 
        else Continue=false
    until | $\pi_i$ |<MaxParentNum and Continue=true
    
```

Fig. 3. The domain-constrained BN structure learning algorithm. D_k denotes the parent domain of k^{th} level, and $level(i)$ means level of x_i and $MaxParentNum$ is a limitation of the number of parents.

The K2 algorithm uses a topological order to maintain the graph as the DAG by maintaining that the prior node cannot be the child of the posterior node without any other DAG checking rules. However, we have to optimize the topological order since a different topological order will have led to a different BN structure. In this paper, we compute the influence score of all the nodes by using mutual information [15], and sorted the topological order with the score. Equations (1) and (2) show the influence score and the mutual information calculation.

$$M_i = \sum_j M(X_i; X_j) \quad (1)$$

$$M(X; Y) = \sum_{x,y} P(x, y) \log \frac{P(x, y)}{P(x)P(y)} \quad (2)$$

3.2 Automatic Modularization of Bayesian Network Structure

The network (G) learned by the structure learning method is divided into several modules by the proposed modularization process as shown in Fig. 4. It contains defining the node domain set based on the module domain and the log data set, and making the arcs of the module BNs based on the network structure (G), and adding the virtual nodes based on the network structure (G).

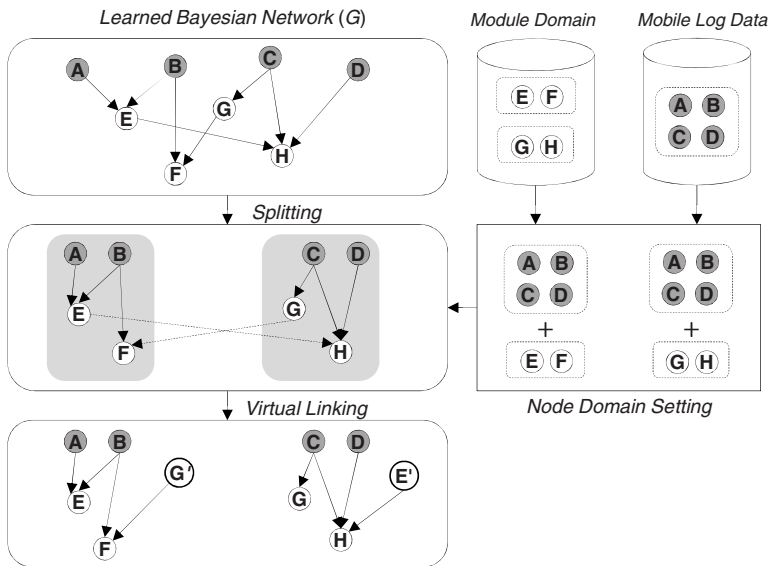


Fig. 4. The modularization procedure from original network into modular BNs

4 Experiments

The log data used in this paper include a GPS log, call log, SMS log, picture log, music playing log, device charging log, and weather log obtained from a website. We collected data from three college students (women) with real-world smart phones for totally 16 days. These users performed subtasks (such as writing activity diaries, shopping, walking and calling) to make the data more substantial. The experimental data was segmented into units of ten minutes. We defined 48 landmarks and used 110 life log contexts (as evidence). To discover the modular BNs, we divided the landmarks into four modules based on four categories (Emotion & status, Everyday life, Events, School life) as shown in Table 2.

Table 2. Module domain definition of landmark nodes

Domain	Landmarks
Emotion & status	bored, busy, cold, concentration, fret, hungry, joy, overflowing joy, sad, sleepy, surprising, throb, tired, troublesome, with leisure, yearning
Everyday life	eat (Chinese), eat (western), eat (Korean), home activity, meet family, moving, ready to go out, ride a vehicle, run, sleeping, supper, using vehicle, walk
Event	date with my date, drinking(alcohol), eat (tee), eat out, hair-cut, meet friend, meet kin, take a walk, traffic jam, weight-training
School life	employment counsel, extracurricular lecture, go to school, late for school, lecture, school activity, school-club activity, study, test in school

4.1 Performance Evaluation of the Discovered BNs

In this section, we describes the test result of the landmark extraction model using 16 days of mobile life log data obtained by the proposed modular BN learning method. We set the *MaxParentNum* parameters (p) as 4 and 8 in the experiment. Fig. 5 shows the results of the landmark reasoning evaluation. Because the number of training data was small, we used the leave-one-out validation method. We compared the monolithic BN and modular BNs with the parameters p=4 and p=8. The computation of the precision rate is $(TP/(TP+FP))$, and the hit rate is $((TP+TN)/(TP+TN+FP+FN))$, where T is true, F is false, P is positive, and N is negative. As shown by the results, the performance of the modular BNs is similar to that of the monolithic BN.

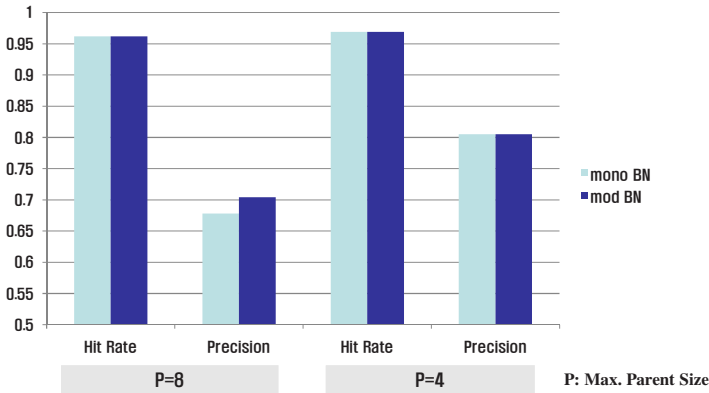


Fig. 5. The comparison of landmark extraction performance

4.2 Complexity Comparison

To compare the complexity for BN inference, we conducted mobile device simulation. Its OS was Pocket PC 2003 and its main memory was 44 MB and it was supported by the Microsoft Pocket PC 2003 SDK toolkit. We tested 10 times of inference running with 20 evidences for each run. Table 3 shows the results. Unfortunately, the mono- BN have not run on the PDA with memory-out-error. The loading of BN file was available but inference was not available since it is too complex. Modular BN works well and it takes 4 seconds.

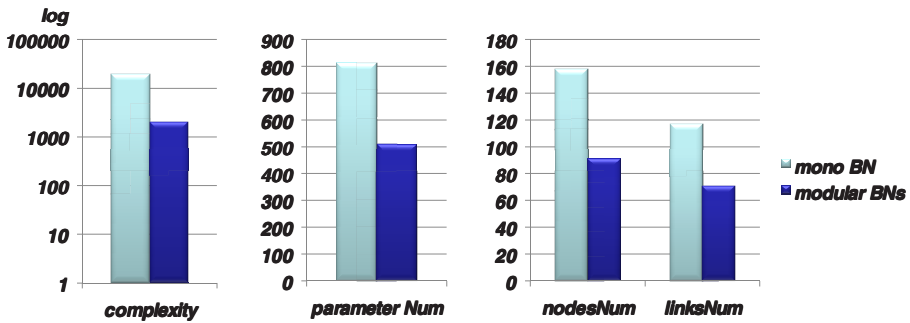
Table 3. The running results on simulation of mobile environment

BN	BN file Loading #	Inference #	Running Time
Mono BN	10	10	Not Available
Modular BNs	30	60	4 sec.

LM is landmark set, L is log context set.

The average number of nodes, parents, and conditional probability values and the level of complexity are calculated by Equation (3), which denotes the simplified time complexity of the exact inference of the BN using the Lauritzen Spiegelhalter (LS) algorithm [16] that is the most popular exact inference algorithm and a junction tree-based algorithm, where n represents the number of nodes, k represents the maximum number of parents for each node used in the LS algorithm [17]. We have replaced the maximum clique size with k , since the clique size is proportional to the parents' size.

$$cmpx_{inf} \cong O(k^3 n^k + kn^2 + 2^k (k+1)n) \quad (3)$$

**Fig. 6.** The modularization procedure from original network into modular BNs

The results show the proposed approach requires less time complexity for expression of probability distribution and inference of landmark.

5 Concluding Remarks

In this paper, we introduced the modularized BN model for efficient operations in mobile environments, and then discovered the modular BNs automatically from the given training data. In experimental results with the real world mobile life log data, we observed that the proposed method was able to reduce the level of complexity.

However, in this paper, we did not sufficiently cover the temporal properties of human landmarks since we did not use a dynamic BN model but only a BN model. In the future, we need to continue research using a dynamic BN model that manages temporal features well. Also, experiments with sufficient real world data should be conducted for a longer period of time.

Acknowledgement

This paper was supported in part by KIST.

References

1. Nokia LifeBlog, <http://www.nokia.com/lifeblog>
2. Schmidt, A., Takaluoma, A., Mäntyjärvi, J.: Context-aware telephony over WAP. *Personal Technologies* 4(4), 225–229 (2000)
3. Lo, B.P.L., Thiemjarus, S., Yang, G.-Z.: Adaptive Bayesian networks for video processing. In: *Int. Conf. on Image Processing*, vol. 1(1), pp. 889–892 (2003)
4. Raento, M., Oulasvirta, A., Petit, R., Toivonen, H.: ContextPhone: A prototyping platform for context-aware mobile applications. *IEEE Pervasive Computing* 4(2), 51–59 (2005)
5. Krause, A., Smailagic, A., Siewiorek, D.P.: Context-aware mobile computing: Learning context-dependent personal preferences from a wearable sensor array. *IEEE Trans. on Mobile Computing* 5(2), 113–127 (2006)
6. Korpipaa, P., Mantjarvi, J., Kela, J., Keranen, H., Malm, E.-J.: Managing context information in mobile devices. *IEEE Pervasive Computing* 2(3), 42–51 (2003)
7. Dourish, P.: What we talk about when we talk about context. *Personal and Ubiquitous Computing* 8(1), 19–30 (2004)
8. Hwang, K.-S., Cho, S.-B.: Modular Bayesian Networks for Inferring Landmarks on Mobile Daily Life. In: *The 19th Australian Joint Conf. on Artificial Intelligence*, pp. 929–933 (2006)
9. Krause, A., Smailagic, A., Siewiorek, D.P.: Context-aware mobile computing: Learning context-dependent personal preferences from a wearable sensor array. *IEEE Trans. on Mobile Computing* 5(2), 113–127 (2006)
10. Horvitz, E., Dumais, S., Koch, P.: Learning predictive models of memory landmarks. In: *CogSci 2004. 26th Annual Meeting of the Cognitive Science Society*, pp. 1–6 (2004)
11. Marengoni, M., Hanson, A., Zilberstein, S., Riseman, E.: Decision making and uncertainty management in a 3D reconstruction system. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 25(7), 852–858 (2003)
12. Tu, H., Allanach, J., Singh, S., Pattipati, K.R., Willett, P.: Information integration via hierarchical and hybrid Bayesian networks. *IEEE Trans. On Systems, Man, and Cybernetics. Part A: Systems and Humans* 36(1), 19–33 (2006)
13. Hwang, K.-S., Cho, S.-B.: Constrained learning method of Bayesian network structure for efficient context classification. In: *Proc. of Korea Information Science Society (In Korean)*, vol. 31(2), pp. 112–114 (2004)
14. Cooper, G.F., Herskovits, E.: A Bayesian method for the induction of probabilistic networks from data. *Machine Learning* 9, 309–347 (1992)
15. Su, J., Zhang, H.: Full Bayesian network classifiers. In: *Proc. of international conference on Machine learning*, pp. 897–904 (2006)
16. Lauritzen, S.L., Spiegelhalter, D.J.: Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society*, 157–224 (1988)
17. Namasivayam, V.K., Prasanna, V.K.: Scalable parallel implementation of exact inference in Bayesian networks. In: *Int. Conf. on Parallel and Distributed Systems (July 2006)*

Skin Pores Detection for Image-Based Skin Analysis

Qian Zhang and TaegKeun Whangbo

Department of Computer Science, Kyungwon University,
Sujung-Gu, Songnam, Kyunggi-Do, Korea
aazhgg@hotmail.com,
tkwhangbo@kyungwon.ac.kr

Abstract. Skin analysis has potential uses in many fields, including computer assisted diagnosis for dermatology, topical drug efficacy testing for the pharmaceutical industry, and quantitative product comparison for cosmetics. In medicine, skin pores are the openings of hair follicles, oil glands, and sweat glands. There are many skin problems associated with skin pores, such as blackheads which are not dirt and cannot be washed away, enlarged pores which are due to over activity of the sebaceous glands in the skin. In computer-aided skin analysis, skin pores are helpful features for skin image registration, skin texture modeling, and skin statement evaluation. In this paper we mainly focus on image-based skin pores detection problem and propose an integrated solution based on fuzzy c-mean algorithm. In our work, research images include images taking by digital camera with long focus lens and images taking by microscope. A global luminance proportion method will be used for skin image preprocessing because of reflection and interreflection of light on the skin surface. We provide experiments to demonstrate the effective and efficiency of our solution.

Keywords: Skin pores detection, Luminance proportion, Fast fuzzy c-mean, Skin Wrinkle, Skin analysis.

1 Introduction

Recently, Skin analysis has been applied in many fields both image-based and modeling-based analysis, involving computer assisted diagnosis for dermatology, topical drug efficacy testing for the pharmaceutical industry, and quantitative product comparison for cosmetics. Quantitative features of skin surface are the significant but difficult task. The skin surface is a complex landscape influenced by view direction and illumination direction [1]. That means we can take skin surface as a type of texture, but this texture is strongly affected by the light and view direction, and even the same skin surface looks totally different. In medical skin research and computer animation, much work has been done in this area [1], [2], [3].

In our great deal of work in skin analysis, such as image registration for cosmetic evaluation, skin blackheads detection, and skin texture modeling, we find the skin pores are the important feature on the skin surface. In the image registration project for cosmetic evaluation, we compare two images from the same skin surface. One is taken without using cosmetic, and the other is taken after four weeks lasted using cosmetic. Effective cosmetic can improve skin surface and wrinkle, which means the

two images for registration are not totally same. According to the medicine, skin pores are inherited, and we cannot change their size, no matter what the cosmetic firms' advertisements. The certain conditions may make pores "appear" larger or smaller, but the position and basis character will be maintained. We choose the same skin pores from the two images and finish the image matching on rotation and translation. In the skin blackheads detection project, we detect skin pores on the nose and evaluate the current skin statement including quantity and area of blackheads. In Our current project 3D skin wrinkle reconstruction, skin pores detection can help us do sparse stereo matching which is the preprocessing for image rectification before dense stereo matching. Other difficulty for skin analysis is acquiring the good data. In our work, we take photos by digital camera with long focus lens for clear skin surface. For the image-based skin evaluation, we make the plaster cast and take photos under microscope.

Image segmentation is the basic operation which separates objects from the background in image. There are many segmentation algorithms using intensity threshold, edge detection and region based approaches, such as Otsu's method, watershed, Gaussian mixture model [4], level set [5], c-means [6], and so on. For skin image, objects are pores and wrinkles which cross each other and form the basic skin structure called grid texture [7]. If we take skin grid as the object in the image, then the background color is close to the object in the skin surface. Also the color has changed by the view direction and light direction.

In this paper, we conclude our previous work and focus on pores detection in the skin image. A preprocessing algorithm for illumination balance is used firstly. In the segmentation part, a fast fuzzy c-means algorithm is applied for initial segmentation, and we set a ratio to separate pores from wrinkles on the skin. The paper is arranged as follows. In Section 2, we introduce our illumination balance algorithm for image preprocessing. The fast fuzzy c-means algorithm is explained in section 3, and also a threshold is set for selection pores from wrinkle. Results for pores segmentation are presented in Section 4 together with a comparison between the proposed and other existing algorithms. Finally, we concluded our work in Section 5.

2 Global Luminance Proportion

Generally skin images can be totally different even by a slight different position because there are influenced highly by light. To solve this problem, we proposed a global luminance proportion algorithm. The method compensates every area of different luminance proportion in order to make the luminance of part area near to that of the whole area.

Suppose an image with size $M \times N$, and the gray level is from 0 to L. We calculate the global average luminance Lum_g . Then we divide image into blocks with size $m \times n$, and calculate the average luminance in each image block Lum_l . For each image block the difference from whole image luminance is decided by $\Delta_{lum} = Lum_l - Lum_g$. It means that in high intensity block the difference is larger than zero; otherwise it is less than zero. In our algorithm, we don't add the global difference to each block directly, but do the interpolation between each block until the

difference matrix size equals to the original image $M \times N$. That is the difference of luminance for each pixel in the image compared with the global average luminance. Combined the difference with original pixel luminance, we get a global luminance proportion image.

Our algorithm is summarized as follows:

1. for image do:
 - a. Calculate average luminance
 - b. Split the image S in V sub-block $S_i^{(1)}, S_i^{(2)}, \dots, S_i^{(V)}$, and calculate average luminance for each sub-block.
 - c. Obtain luminance difference matrix D
2. Interpolation algorithm for matrix D until element number in matrix equals to $M \times N$
3. Merge matrix D and original image S into new image with size $M \times N$

3 Image Segmentation

In our research, we want to separate the skin grid structure including pores and wrinkles first. Then we select pores from wrinkle through setting a threshold. The previous work has been done in paper [7]. They perform an energy transformation to map the pixels from grayscale space into energy space. The filtering can be manually turned to get a different result. Normal curvature of the energy surface is utilized to identify the principal direction and the ridge centerlines can be detected at the image locations where the principal direction is perpendicular to the normal vector. Through energy transform, both pores and wrinkles can be emphasized. For pores detection, we don't need to detect wrinkles, and pixel luminance intensity of pores is not larger than the wrinkles'. We apply an image segmentation algorithm in skin image directly. In image registration system under microscope, the process should be completed fast, and high accuracy also is required. In other automatic systems, such as blackheads detection on the nose, skin texture modeling, the same requirements are proposed. In general, the fuzzy c -means approach is effective for image segmentation automatically. For pores detection, through reducing data store space and simply the object function, we accelerate the fuzzy c -means approach for initial image segmentation.

3.1 The Fast Fuzzy C-Means Algorithm

Fuzzy c -mean (FCM) is an unsupervised clustering algorithm that has solved many problems successfully including feature analysis, clustering and classifier design. A generalization of the FCM algorithm was proposed by Bezdek [1981] through a family of objective functions. For image segmentation, it is better than the hard c -means algorithm at avoiding local minima; FCM can still converge to local minima of the squared error criterion [8]. On the other hand, FCM is suit for data clustering in the high dimensional feature space.

The FCM minimize a weighted squared error criterion function based on fuzzy criterion. It is defined as follows:

$$J_m(U, V, X) = \sum_{i=1}^c \sum_{k=1}^n u_{ik}^m \|x_k - v_i\|_A^2, \quad 1 < m < \infty \tag{1}$$

Where $V = (v_1, v_2, \dots, v_c)$ is a vector of unknown cluster centers $v_i \in R^p$. The value of u_{ik} represents the grades of membership of pixel data x_k of set $X = (x_1, x_2, \dots, x_n)$ to the i th cluster. The inner product of defined by a norm matrix A is a measure of similarity between a pixel data and the cluster centers. A nondegenerate fuzzy c -partition of X is conveniently represented by a matrix $U = [u_{ik}]$.

For all i and k , if $\|x_k - v_i\|_A > 0$, then (U, V) may minimize J_m only, when $m > 1$ and J_m can be minimized by iteration approach, for the $(b+1)$ th iteration,

$$v_i^{(b+1)} = \frac{\sum_{k=1}^n (u_{ik}^{(b)})^m x_k}{\sum_{k=1}^n (u_{ik}^{(b)})^m} \quad \text{for } 1 \leq i \leq c, \tag{2}$$

$$u_{ik}^{(b+1)} = \frac{1}{\sum_{j=1}^c \left(\frac{\|x_k - v_i^{(b)}\|_A^2}{\|x_k - v_j^{(b)}\|_A^2} \right)^{1/m-1}}, \quad \forall i, j \text{ for } 1 \leq i \leq c, 1 \leq k \leq n. \tag{3}$$

If $\|V^{(b+1)} - V^b\| < \epsilon$, iteration stops, otherwise it continues. Where ϵ is the threshold we set to determine the processing termination.

In fact, FCM is the ultimate method of solving the problem of non-convex optimization iterative algorithm. It's the algorithm with high time consuming, especially for the high resolution image. We are inspired in formula (2), in which only the calculated sum of different data is necessary not the each data. After we obtain the required sum of data, we do not need to store the individual data. This helps us to reduce the data store space. Another inspiration comes from formula (1) through experiment. The linear distance is given by the equation

$$J_m(U, V, X) = \sum_{i=1}^c \sum_{k=1}^n u_{ik}^m \|x_k - v_i\|_A, \quad 1 < m < \infty \tag{4}$$

It produces almost the same results in practice, with the clustering center values identical to those from the least square method. In our research, linear distance measure is chosen to simplify the computation.

3.2 Pores Detection

After the initial image segmentation, pixels in image are labeled to 8-connectivity. In this part we classify the skin pores and wrinkles from 8-connectivity labeled image. At first we compute the quadric moment for classifying pores and wrinkles. After that, we separate skin pores by calculating the ration between row and column moments.

Step 1, Computing moments at each region

The moment of row and column is defined in the following:

$$\begin{aligned}\mu_1 &= \frac{1}{p} \sum_{i=1}^m (x_i - m_c)^2 \\ \mu_2 &= \frac{1}{q} \sum_{j=1}^n (y_j - m_r)^2\end{aligned}\tag{5}$$

Where $p = \sum_{i \in Label} 1$, $q = \sum_{j \in Label} 1$, $(x, y) = \{(x, y) | (x, y) \in Label\}$, and (x, y) de-

scribes the pixel in label area position. m_c, m_r are the mean of column and row value separately.

Step2, Pores and wrinkles classification by calculating ratio

We calculate the ratio through formula (6) as follows, after acquiring the value of moments from step 1.

$$ratio = \frac{\mu_1}{\mu_2} \times 100\%\tag{6}$$

The threshold for ratio is decided by the experience. For each region labeled if ratio lies in 0.1 and 6.5, it means the pore; otherwise, it is the wrinkle, and we will remove it.

4 Experiment

To illustrate the validity of the solution proposed in the paper, we use threshold segmenting method (TS), fuzzy based histogram algorithm (FTS) and our proposed method (FFCM) to detect the pores in our test images respectively, and make a comparison between them. In our experiment, we have two type images, one type taken by digital camera with long focus lens involving the nose image for blackheads detection; another type taken by microscope and the image from the plaster cast which is the sampling on the skin surface. Our system performs on the personal computer with Intel Core 2Duo 2.66GHz CPU and 2G memory.

For luminance proportion part, we use bicubic [9] interpolation algorithm for luminance difference matrix with the same size with original image. In threshold segment method, we calculate the maximum and minimum grayscale value in the image, and get the global threshold by average the two values. Actually, it is an old but efficient segmentation method. The fuzzy based histogram method was proposed in paper [10]. In this method, the threshold is decided through index of fuzziness and image entropy. It is another effective algorithm for global segmentation. In the FCM

method, $\varepsilon = 0.00001$ our experiment results are shown in figure 1 and figure 2 in the following. For the nose image, only the pores are on the skin surface. We don't need separate pores from wrinkle through a ratio. For the skin surface under microscope, the post-processing of pores and wrinkles classification is necessary. Table 1 gives the time consuming comparison of the different segmentation approaches for nose image (size 512×512) in figure 1 image B. The values given in the list are calculated average values from several experiments.

For the skin image, we do the post-processing for pores and wrinkles classification. In our research, we want to detect the pores not skin grid structure. The pores detection result is shown in figure 3. Compared with the original image, there are some lost pores in our final result. Except professional pores detection like the nose blackheads measurement, we don't need to detect all the pores on skin image exactly. Because pores detection is not the main research purpose for image registration in cosmetic evaluation and skin texture modeling. The purpose that we focus on pores detection is to select skin features for skin analysis and skin texture modeling. In fact, it is a hard problem especially to deal with the real skin image without a skin sampling plaster cast under microscope.

Table 1. Performance comparisons

Performance	TS	FTS	FCM	FFCM
CPU time (s)	0.0313	0.1719	16.5274	5.6875

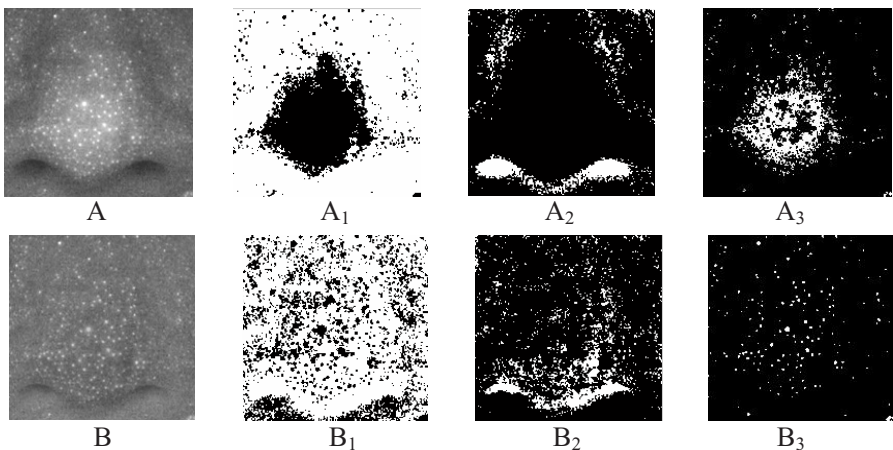


Fig. 1. The results are from nose image for blackheads detection. Among them A is the original image, and A₁, A₂, A₃ are the detection results through TS, FTS, and FFCM algorithm respectively based on image A before luminance proportion. B is the original image after luminance proportion, and B₁, B₂, B₃ are the detection results through TS, FTS, and FFCM algorithm respectively based on image B.

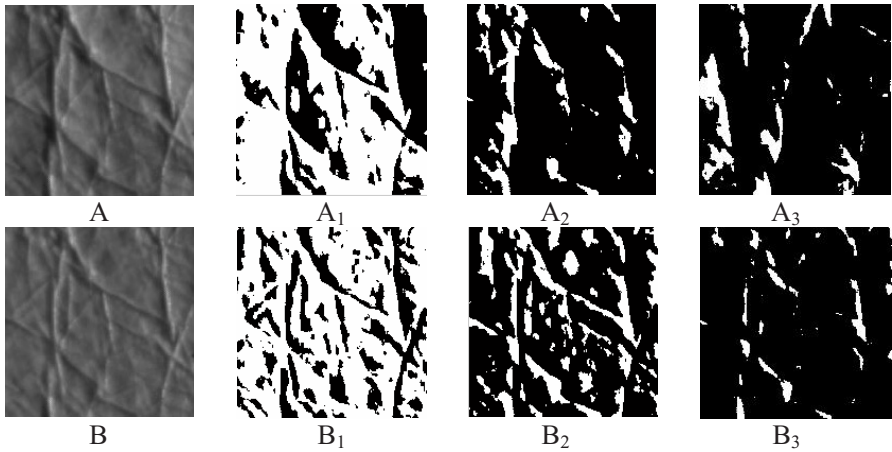


Fig. 2. The results are from skin image for image registration. Among them A is the original image, and A₁, A₂, A₃ are the detection results through TS, FTS, and FFCM algorithm respectively based on image A before luminance proportion. B is the original image after luminance proportion, and B₁, B₂, B₃ are the detection results through TS, FTS, and FFCM algorithm respectively based on image B.

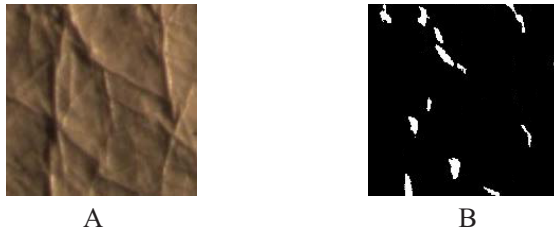


Fig. 3. The pores detection result. A is the original skin image and B shows the pores detection result.

5 Conclusion

Skin analysis is a meaningful research for both image-based skin texture and modeling based skin surface. Through a number of projects related with skin analysis, we find the pores detection is very significant for skin measurement, skin feature extraction and 3D skin modeling. In the paper, we proposed a solution for pores detection in which the luminance proportion algorithm is very useful to deal with the problem caused by light influence, and FCM algorithm is effective for skin image segmentation. The skin pores detection is useful for skin analysis feature selection, although it is a hard problem because of the skin surface influenced by view direction and luminance direction strongly.

References

1. Cula, O.-G., Dana, K.-J., Murphy, F.-P., Rao, B.-K.: Skin Texture Modeling. *International Journal of Computer Vision* 62(1-2), 97–119 (2005)
2. Cula, O.-G., Dana, K.-J.: Image-based skin analysis. In: *The 2nd International Workshop on Texture Analysis and Synthesis*, pp. 35–40 (2002)
3. Dana, K.J., Nayar, S.K.: Histogram Model for 3D Textures. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 18–624 (1998)
4. Freifeld, O., Greenspan, H., Goldberger, J.: Lesion detection in noisy MR brain images using constrained GMM and Active Contours. In: *Proceedings of IEEE International Symposium on Biomedical Imaging*, pp. 596–599 (2007)
5. Malladi, R., Sethian, J.A., Vemuri, B.: Shape Modeling with Front Propagation: A Level Set Approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 17(2), 18–175 (1995)
6. Kwok, T., Smith, R., Lozano, S., Taniar, D.: Parallel fuzzy c-means clustering for large data sets. In: Monien, B., Feldmann, R.L. (eds.) *Euro-Par 2002. LNCS*, vol. 2400, pp. 365–374. Springer, Heidelberg (2002)
7. Zhu, L.-G., Qin, S.-Y., Zhou, F.-G.: Skin image segmentation based on energy transformation. *Journal of Biomedical Optics* 9(2), 362–366 (2004)
8. Ma, L., Staunton, R.-C.: A modified fuzzy C-means image segmentation algorithm for use with uneven illumination patterns. *Pattern Recognition* 40(11), 3005–3011 (2007)
9. Durand, C.-X., Faguy, D.: Rational zoom of bit maps using B-spline interpolation in computerized 2 - D animation. *Computer Graphics Forum*, 27–37 (1990)
10. Pal, S.-K., King, R.-A., Hashim, A.-A.: Automatic gray level thresholding through index of fuzziness and entropy. *Pattern Recognition Letters* 1(3), 141–146 (1983)

An Empirical Research on Extracting Relations from Wikipedia Text

Jin-Xia Huang, Pum-Mo Ryu, and Key-Sun Choi

SWRC, Computer Science Division, EECS Dept. KAIST
335 Gwahangno, Yuseong-gu, Daejeon, 305-701, Republic of Korea
{hgh, pmryu, kschoi}@world.kaist.ac.kr

Abstract. A feature based relation classification approach is presented, in which probabilistic and semantic relatedness features between patterns and relation types are employed with other linguistic information. The importance of each feature set is evaluated with Chi-square estimator, and the experiments show that, the relatedness features have big impact on the relation classification performance. A series of experiments are also performed to evaluate the different machine learning approaches on relation classification, among which Bayesian outperformed other approaches including Support Vector Machine (SVM).

Keywords: Information extraction, relation classification, feature-based, relatedness information.

1 Introduction

Extracting relationships between entities from text is one of the most challenging issues in information extraction. The task of relation extraction is identifying relationships between two or more entities in given context. Feature-based relation extraction, which has been broadly employed in these years [1-3], investigates various features including lexicon, part-of-speech (POS) information, syntactic information and semantic knowledge to represent relation candidates, and classifies the relations with diverse classifiers.

In this paper, aiming at building domain ontology from texts, the problem of intra-sentence relation extraction is dealt with. The relation candidates are detected from Wikipedia texts with lexical patterns, and classified into certain relation types which are predefined for IT domain. The well known features, including word, POS and syntactic information [1-3], are employed with the relatedness features between patterns and relation types which are proposed in this paper. The relatedness information is acquired from WordNet [4] – which is semantic relatedness information; and from training corpus – which is probabilistic relatedness information. The experiments in this paper show that the relatedness information contributes to the classification performance in a significant way.

The evaluation on feature impact is also an important issue in feature based relation classification. To adopt the features in the order of their importance and avoid employing noisy features, we evaluated the impacts of all features by using Chi-square estimator [5]. The experiments show that, the relatedness features have big impact on the relation classification performance.

The contributions of this paper are as following: 1) semantic and probabilistic relatedness features are proposed for feature-based relation classification; 2) the importance of each feature set is evaluated, so that the features can be adopted gradually based on their importance; 3) a series experiments are performed to evaluate the different machine learning approaches on relation classification task.

The rest of the paper is organized as follows: Section 2 gives the problem definition. Section 3 presents how the relation types and patterns are selected, with a description on entity detection and pattern-based relation detection. Section 4 presents in detail the feature sets which are employed in this paper. Section 5 describes the experimental evaluation, and draw to conclusion in Section 6.

2 Problem Description

The research problem of this paper is extraction of explicit relationships between entities occur in a sentence. The entities can be domain specific terms, noun phrases, and named entities. The entities and the relation candidates are detected by a pattern matching approach, and then classified to certain relation types. Given entities e_1 and e_2 occur in context W , which matches given pattern p . What we want to predict is its relation type r :

$$f:(e_1, e_2, p, W) \rightarrow r$$

The relation candidates are represented with features, and then put into the relation classifier to predict its relation type r . The relation classifier is trained with labeled data, which already verified by human annotators.

The relation type r can be one of *isa*, *usedFor*, *produces*, *provides*, and no-relation. Considering each relation type already has its own patterns predefined, the multi-classification task can be transferred into a binary classification task, in which the predicted result (classification category) is either *yes* or *no* for certain relation type r .

3 Pattern-Based Relation Detection

3.1 Relation Types and Patterns

The four relation types in this paper are selected according to their frequencies in IT domain:

- *isa*: a *subclass* or *instanceOf* relation between two classes or an instance and a class.
- *usedFor*: in a relation of “A *usedFor* B”, domain A can be used for, or used in B.
- *produces*: range is generated, created, or manufactured by domain.
- *provides*: range is offered, provided, or supported by domain.

Not only the relation types, but also the lexical patterns are predefined for each relation type. For example, *isa* relation has patterns “be, be a form of, such as”, *provides* has patterns “provide, offer, invest”, *produces* has patterns “produce, invent, establish”, *usedFor* has patterns “be used for, be used as, be available for”. In practice, the

patterns consist of a sequence of word and POS pairs with syntactic dependency relations among the words. Figure 1 shows an example of dependency structure for a sentence “*GMail is a special kind of free web-based e-mail*”, which matches a predefined *isa* pattern. In the pattern, noun phrases annotated as *Domain* and *Range* match to entities in text to identify hypernym and hyponym of *isa* relation. This pattern can capture potentially relevant linguistic relations which are hardly found in simple pattern matching method.

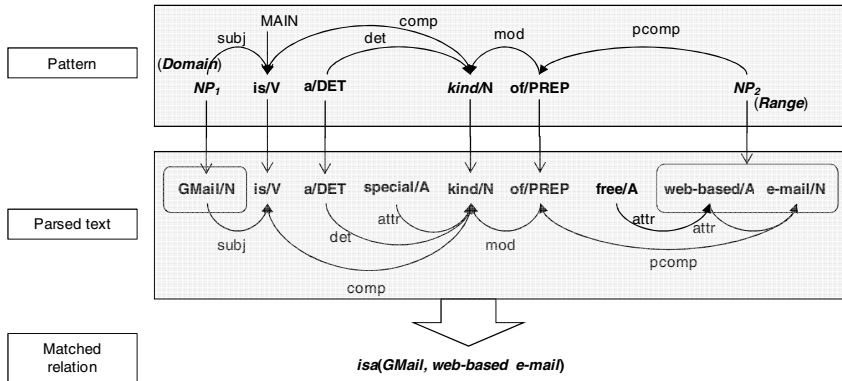


Fig. 1. A pattern, parsed text and matched isa relation

3.2 Pattern Matching

We analyze dependency structure of Wikipedia text to apply the defined dependency based patterns. For the linguistic analysis, we use Connexor syntactic dependency parser [6], which provides dependency structure along with grammatical function assignment, part-of-speech and lemmatization (Figure 1). After the linguistic analysis, we recognize entities in sentences. The entities are matched Domain or Range nodes of patterns and become two arguments in relation candidates. Nouns or noun phrases equal to Wikipedia page names are recognized as entities. In Figure 1, “GMail” and “web-based e-mail” are selected as entities.

The analyzed sentences are matched to the list of patterns sequentially. For a given pattern, starting from main node, its child nodes are recursively matched to the sentence structure. In Figure 1, once the main node “is/V” is found in a sentence structure, its two nodes and their links, (is/V ← subj ← NP₁) and (is/V ← comp ← kind/N), are checked again. Referring expression and negative expression are not allowed in matching process although their syntactic structure matches to a pattern. Referring modifiers, such as “the”, “his” and “this”, refer other entities within the same context, and whose meaning depends on the context where it is used. Modifiers such as “not”, “no” and “neither” contribute to negative relations. Structure based pattern matching enables to extract long distance relations by modifiers or nested phrases.

4 Feature-Based Relation Classification

Select what kind of features has strong impact on the classification performance. The features computed in this paper are described below, with an example of parse tree given in Figure 2 for a sentence “**Application streaming is a relatively new form of software distribute method** using application virtualization”. The relation candidate (application streaming, software distribute method) is extracted with an *isa* pattern “be a * of”.

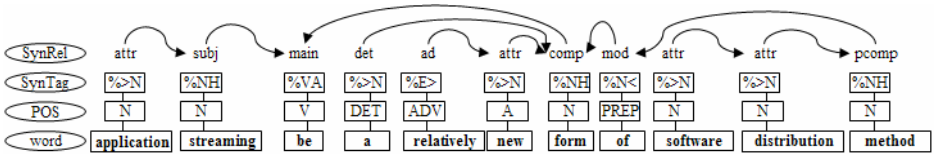


Fig. 2. The parsing results on given example

Word features: the most basic features the relation candidate has. It includes the string which match the pattern (PAT_be_a_relatively_form_of), the main word of the pattern (PAT_be), the domain and range entities of the relation candidate (DOM_application_streaming, RAN_software_distribution_method), the headwords of the entities (WH1_streaming, WH2_method), and all words of the two entities (WM1_application, WM1_streaming, WM2_software, WM2_distribution, WM2_method).

Context features in word level: the words after the domain entity (WA#) and before the range entity (WB#) in the parse tree. # can be 1 or 2, means the position of the words in the context: 1 is right before or after the entity, 2 is the other one (WB1_of, WB2_form, WA1_be, WA2_a). It is also a word level feature.

POS features: POS tag of all above word level features (PM1_N, PM1_N; PM2_N, PM2_N, PM2_N; PB1_PREP, PB1_N; PA1_V, PA2_DET).

Syntactic features: syntactic tags of all above word level features (TM1_>N, TM1_NH; TM2_>N, TM2_>N, TM2_NH; TB1_N<, TB2_NH; TA1_VA, TA2_>N).

Dependency features: dependency tags of all above word level features (RM1_attr, RM1_subj; RM2_attr, RM2_attr, RM2_pcomp; RB1_mod, RB2_comp, RA1_main, RA2_det).

Relatedness features: the probabilistic relatedness information between the pattern and the relation type (PATProb:0.7), the probabilistic and semantic relatedness information between the main word of the pattern and the relation type (PATMainProb:0.5, PATSim:1).

Probabilistic relatedness information is acquired from labeled data, by calculating the percentage of positive cases of the patterns (or main words of the patterns) in the relation type. Actually it is the accuracy of the patterns shown in pattern matching

procedure. For example, the pattern “be a form of” has 71.87% of accuracy (PAT-Prob:0.7), and the patterns which have “be” as their main words have accuracy 53.02% in average (PATMainProb:0.5).

The semantic relatedness between the main word “be” and the relation type *isa* is 1 (PATSim:1), it is acquired from WordNet. For certain relation type, collect the main words of its patterns $\{w_1, \dots, w_i, \dots, w_n\}$, for example, $\{\text{use, employ, available}\}$ for relation type *usedFor*, the semantic relatedness between the main word w_i and the relation type r , is related to how many semantically closed words employed for the relation type. The more similar words of w_i employed in the patterns for the relation type r , the higher relatedness score w_i gains for r (Equation 1). The relatedness score w_i is normalized by the maximum score among all main words $\{w_1, \dots, w_i, \dots, w_n\}$, which means, the relatedness score is a positive decimal less than 1 (Equation 2).

Let $dis(w_i, w_j)$ indicate the distance between w_i and w_j in WordNet, $sim(w_i)$ be the semantic relatedness of w_i , we have:

$$score(w_i) = \sum_{j=1}^n 1 / dis(w_i, w_j) \quad (1)$$

$$sim(w_i) = score(w_i) / \{\max_{j=1}^n score(w_j)\} \quad (2)$$

In equation (1), the distance of the words in the same synset is 1, the one of direct hyponym and hypernym is 2, and it is infinity if there is no path between two words in WordNet.

To given example $\{\text{use, employ, available}\}$ for relation type *usedFor*, $score(\text{use})$ and $score(\text{employ})$ are both 2, while $score(\text{available})$ is 1, because $dis(\text{use, employ})=1$ (these two words are in the same synset in WordNet), and $dis(\text{available, available})=1$ too. According to equation (2), the final semantic relatedness $sim(\text{use})$ is 1, while $sim(\text{available})$ is 0.5.

5 Experiments

5.1 Evaluation Method

Four different possible outcomes of a single prediction are described in Table 1. The *true positive* and *true negative* are correct classifications. A *false positive* is when the outcome is incorrectly predicted as *yes* (or *positive*), when it is in fact *no* (*negative*). A *false negative* is when the outcome is incorrectly predicted as *negative* when it is in fact *positive*. Precision, recall and F-measure are evaluated for positive cases of relation r ; while accuracy is evaluated for both cases.

Table 1. Different outcomes of binary prediction

		Predicted results	
		Yes	No
Actual results	Yes	True positive (a)	False negative (b)
	No	False positive (c)	True negative (d)

$$P = \frac{a}{a+c} \quad R = \frac{a}{a+b} \quad F = \frac{2P \times R}{(P+R)} \quad A = \frac{a+d}{a+b+c+d} \quad (3)$$

5.2 Dataset and Performance

Wikipedia pages in IT domain are downloaded for the experiments. The relation candidates are extracted from the first sections of the pages, which normally are definitions and core descriptions, by matching predefined patterns on parsed texts. Connexor parser [6] is used for parsing, a machine learning software WEKA [7] and SVM toolkit LibSVM [8] is adopted for relation classification.

Table 2. Dataset

Relation type	Pattern number	Training set (positive cases)	Test set (positive cases)
<i>isa</i>	89	35,389 (54.7%)	1,158 (50.2%)
<i>usedFor</i>	22	720 (43.2%)	126 (42.9%)
<i>produces</i>	46	1,038 (51.4%)	155 (38.1%)
<i>provides</i>	17	1,803 (48.2%)	317 (47.3%)

The data set used in the evaluation is as Table 2: the first column is the relation type; the second column shows the number of patterns which are adopted in the relation candidate extraction; the relation candidates which already verified by human annotators are separated to training set in third column and test set in forth column, where the percentages of positive cases show how many of the candidates are really hold the relation type - it is the accuracy of pattern matching module indeed, and can be considered as baseline of the relation classification system.

The evaluation results on the different relation types are given by Bayesian classifier in WEKA [7] (Table 3).

Table 3. Performances on different relation types

Relation type	Accuracy	Precision	Recall	F-measure
<i>isa</i>	74.7%	70.5%	82.5%	76.0%
<i>usedFor</i>	61.2%	54.0%	69.0%	60.6%
<i>produces</i>	71.0%	67.8%	68.8%	68.3%
<i>provides</i>	64.7%	63.0%	64.4%	63.7%

5.3 Evaluation on Feature Selection

For the evaluation of features, we evaluated the impact of all features using Chi-square estimator [5] which is provided by WEKA [7]. The Chi-square evaluates features individually with respect to the classification categories in training data. The average ranking of each feature is as following: word and context feature sets are on the top, then relatedness feature set is next, with POS feature set, dependency feature set, and syntactic feature set follow next.

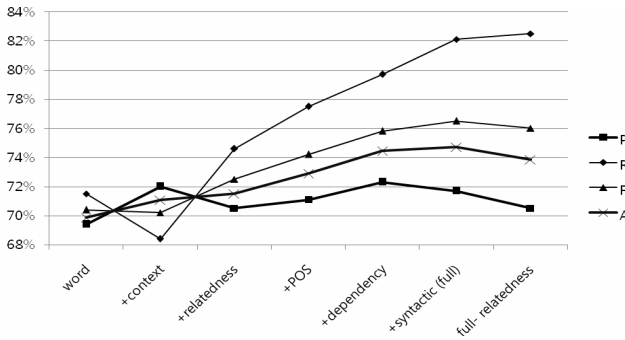


Fig. 3. The contribution of feature sets for *isa* relation classification (Bayesian)

To evaluate the contribution of each feature set, we conducted iterative *isa* relation classification experiments with Bayesian classifier. New feature set is increasingly added in the order of Chi-square ranks. From Figure 3, we can see that as the feature sets are added, both accuracy and F-measure also increased. Although precision do not significantly changed, recall increases significantly as features are added. This result indicates that we can detect correct relations using simple word-level information such as ‘word’ and ‘context’ feature sets; however, to extract more relations (to increase recall), we need to apply additional information because the features generalize conditions for relation extraction.

5.4 Evaluation on Different Machine Learning Approaches

The performances of Bayesian classifier, Naïve Bayesian classifier, Nearest Neighbor classifier, decision tree, and SVM are evaluated in this section. The prior four classifiers are provided by WEKA [7] and SVM is provide from LibSVM [8].

Bayesian classifier and Naïve Bayesian classifier are simple probabilistic algorithms which apply ‘Bayes’ theorem’. The term ‘Naïve’ is because it is based on the assumption that the attributes on the training samples are independent and there is no hidden or latent attributes [9]. Instance based learning is a non-parametric inductive learning paradigm that stores training instances in a memory on which predictions of new instances are based. In our experiment, we used IB1 [10], a NN classifier that uses Euclidian distance metric. Decision tree is a top-down induction method. C4.5 is the advanced version of decision tree algorithm [11]. SVM, based on a statistical learning theory, starts to learn the data in the high dimensional feature space, in the learning phase, by minimizing the magnitude of the weight vector constrained by the separation into an unconstrained problem with the help of multiplier parameter. In this stage, SVM extracts the support vectors only. Based on the support vectors information, SVM produces the final output function in the decision phase [12].

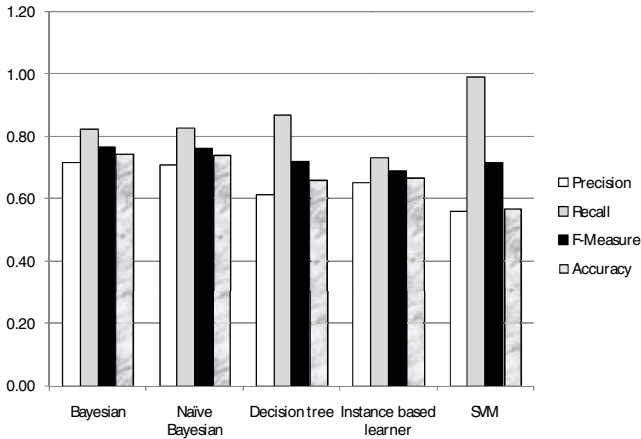


Fig. 4. Comparison of different relation classifiers for *isa* relation

As Figure 4 shows, Bayesian classifier shows the best F-measure in the evaluation, although the recall of the SVM is the highest one.

6 Conclusion

In this paper, a feature-based approach for relation classification is presented. Both probabilistic and semantic relatedness information between patterns and relation types is employed as features. The probabilistic relatedness information can be acquired from training data, while the semantic relatedness can be calculated using WordNet or other similar taxonomies. The experiments on feature-selection with Chi-square estimator showed that the relatedness feature set has high impact in the relation classification, and even outperforms the impact of POS, syntactic or dependency feature set. A series of experiments is also performed to evaluate the different machine learning approaches, including Bayesian, Naïve Bayesian, NN, Decision tree, and SVM. The evaluation result shows that the Bayesian classifier outperforms other classifiers.

As the future work, we are focusing on how to use unlabeled data in an efficient way for a large scale task – extract relations from web scale texts. In the meanwhile, with noticing that the performance given in this paper is still not good enough for practical using in the ontology building field, we are also exploring relatedness information between the entity terms and the relation types, while this paper only focus on the pattern related relatedness.

References

1. Kambhatla, N.: Combining Lexical, Syntactic, and Semantic Features with Maximum Entropy Models for Extracting Relations. In: Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (2004)
2. Zhou, G., Su, J., Zhang, J., Zhang, M.: Exploring Various Knowledge in Relation Extraction. In: Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics, pp. 427–434 (2005)

3. Zhou, G., Zhang, M.: Extracting relation information from text documents by exploring various types of knowledge. *Inf. Process. Manage.* 43(4), 969–982 (2007)
4. Miller, G.A.: WordNet: An online lexical database. *International Journal of Lexicography* 3(4), 235–312 (1990)
5. Manning, et al.: Text classification and Naïve Bayes. In: *An Introduction to Information Retrieval*, pp. 253–287. Cambridge University Press, Cambridge (2008) (online version)
6. Connexor: The Connexor Language Parsers and Taggers for English Website (2008), <http://www.connexor.eu/>
7. Witten, I.H., Frank, E.: *Data Mining: Practical machine learning tools and techniques*, 2nd edn. Morgan Kaufmann, San Francisco (2005)
8. LIBSVM, A Library for Support Vector Machines (2008), <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>
9. John, G.H., Langley, P.: Estimating continuous distributions in Bayesian classifiers. In: *Proceeding of the 11th conference on Uncertainty in Artificial Intelligence*, pp. 338–345. Morgan Kaufmann, San Mateo (1995)
10. Aha, D., Kibler, D.: Instance-based Learning Algorithms. *Machine Learning* 6, 37–66 (1991)
11. Quinlan, R.: *C4.5: Programs for Machine Learning*. Morgan Kaufman, San Mateo (1993)
12. Vapnik, V.N.: An overview of statistical learning theory. *IEEE Trans. Neural Network* 10(5) (1999)

A Data Perturbation Method by Field Rotation and Binning by Averages Strategy for Privacy Preservation

Mohammad Ali Kadampur and Somayajulu D.V.L.N.

Department of Computer Science and Engineering
National Institute of Technology Warangal-506004, A.P. India
ali.kadampur@gmail, soma@nitw.ac.in
www.nitw.ac.in

Abstract. In this paper a novel technique useful to guarantee privacy of sensitive data with specific focus on numeric databases is presented. It is noticed that analysts and decision makers are interested in summary values of the data rather than the actual values. The proposed method considers that the maximum information lies in association of attributes rather than their actual proper values. Therefore it is aimed to perturb attribute associations in a controlled way, by shifting the data values of specific columns by rotating fields. The number of rotations is determined via using a support function for association rule handling and an algorithm that computes the best-choice rotation dynamically. Final summary statistics such as average, standard deviation of the numeric data are preserved by making bin average replacements for the actual values. The methods are tested on selected datasets and results are reported.

1 Introduction

Privacy is defined as “freedom from unauthorized intrusion” [15]. It is a deterrent against individually identifiable data in the process of knowledge extraction. Data mining technology is used for extracting knowledge from vast quantities of data. However the use of this technology has raised the concern that individual privacy is violated. Therefore the data mining technique must ensure that any information disclosed

1. cannot be traced to an individual; or
2. does not constitute an intrusion.

There are multiple approaches to achieve these goals[15]. Data perturbation is one of the methods for preserving privacy[2][12][15]. In perturbed data bases, if unauthorized data is accessed, the true value is not disclosed. Data perturbation techniques in effect distort the data in different ways before presenting it to the data mining algorithm, thus individually identifiable (private) values are not revealed. The privacy-preserving properties of such databases are a result of the perturbation. In this paper a composite novel method for data perturbation is proposed.

2 Related Work

In order to distort the data and preserve individual privacy, researchers have employed methods such as data encryption[11][13], Data randomization[12][15], Data swapping

[1],Data anonymization[4][6][7],Geometric transformation[2][11] and Nearest Neighbor Data Substitution (NeNDS)[11].The Fast Fourier Transform (FFT) and Wavelet transformation based data perturbation methods also have been reported [20] .

3 Motivation

Motivation for our approach is the observation that maximum information lies in the association of attributes (tuples) rather than the attribute value proper. Therefore it is proposed to break this association of attributes in a controlled and well recorded manner in order to allow only the legitimate users to access the original data. The integrity of the horizontals in the table is broken by shifting the data values of specific columns(fields) by rotating the fields. The number of rotations to be performed is evaluated by considering the internal associations of the data.

Illustration : Consider the following two matrix instances

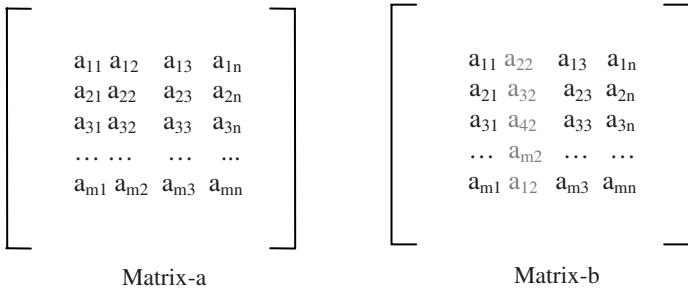


Fig. 1. matrix-a original table, matrix-b perturbed table

Let a_{ij} indicate the atomic value in the i^{th} row and j^{th} column. Let N be an integer indicating number of rotations and R be the number of tuples in the table(matrix). In our approach we try to perturb the i^{th} row by rotating j^{th} column by N times. The column “j” will be chosen depending upon the confidentiality associated with it. The Number of rotations N on j^{th} column is computed as a function of S, the support. $N=f(\text{support})$. The new value of data in the i^{th} row after perturbation in j^{th} column gets changed depending on N and R values in the table. The new perturbed value will be obtained by

$$a_{ij} = a_{i (j+N) \bmod R} \tag{1}$$

Choice of number of rotations N on the field is critical to the method of field rotations. Proper value of N is computed by finding association rules and their support values.

3.1 Association Rule

If $I=\{i_1,i_2,i_3,\dots,i_m\}$ is an item set then an association rule is an implication of the form $X \Rightarrow Y$, where $X \subset I$, $Y \subset I$ and $X \cap Y = \phi$ is true[5][15]. The support S is a number

indicating percentage of transactions containing $X \cup Y$ in the total number of transactions in a data base. Therefore support S is formally defined as in equation-2 below.

$$S = |X \cup Y| / |T_n| \tag{2}$$

T_n = Total number of transactions in a data base D.

3.2 Our Approach

In our first approach we mine the data base and find all the association rules that satisfy minimum support criteria. We then choose a new value of support which is near to the actual support value of the association rule.

Let S_a be the actual support and S_k be the chosen approximate support nearest to S_a for an association rule. $S_k = (k \times S_a)$ where k is a fraction $1 > k > \text{min_support}$; Suppose $I = \{A, B, C, D, E, F\}$ is an item set and $X \Rightarrow Y$ is the only association rule which meets the minimum support criteria. If support for this rule is 60% then as per the above notations $S_a = 60$ and with $k = 0.98$ (assumed), $S_k = 58.8 \sim 59$; We also note that $X \subset I$ and $Y \subset I$. If $BC \Rightarrow ADF$ is the rule then we try to rotate the fields contributing each of B, C, A, D, F such that their new support is adjusted to 59. our algorithm computes the best possible rotation number N for each column that is participating in association rule. $N = f(\text{support})$. It is to be recalled that even after perturbation the summary statistics for association rule mining have to remain same, this is achieved here by making an approximation on S_k and computing the N values for field rotations. The following Table-1 illustrates how our algorithm converges on N value. It shows $N_{[\text{array}]}$ vectors being formed for attributes X and Y in the association rule $X \Rightarrow Y$.

Maximum rotations that one can perform on any table is equal to the number of tuples R in the table. The Table-1 shows support value listing for all possible rotations (1 to R) in Y attribute for every single rotation in X attribute ($N \times 1$ denotes one rotation in X attribute, $N \times 2$ denotes 2 rotations in X attribute etc). Suppose that S_{64} is the nearest best support, then it indicates 6 rotations on X attribute and 4 rotations on Y attribute.

Table 1. $N_{[\text{array}]}$ vectors being formed for attributes X and Y

<i>Number of rotations in X attribute</i>	<i>Number of rotations in Y attribute</i>	<i>Calculated support for $X \rightarrow Y$ in each set of rotations for X and Y</i>
$N \times 1$	$(N_y1 \dots N_yR)$	$(S_{11} \dots S_{1R})$
...	$(N_y1 \dots N_yR)$...
$N \times R$	$(N_y1 \dots N_yR)$	$(S_{R1} \dots S_{RR})$

3.3 The Metadata Structure

In the process of perturbation by field rotations , our algorithms will optimize the number of rotations required on each field and return such statistics on field rotation, these statistics are preserved for any future recovery of the original dataset. This metadata structure is the key for original data recovery and this table will be stored at a secured place by the data owner. The sample metadata structure is shown in table-2. The integers in the table-2 indicate the number of rotations.

Table 2. Meta data structure showing number rotations on each field in a table

Table	Field1	Field2	Field3	Field4	..	fieldN
T1	2	4	25	7	..	8
T2	3	8	6	1	..	15
T3

3.4 Numeric Data Perturbation

Numeric data is perturbed by replacing the proper values by bin averages see table-3. A bin is a prespecified set of records.

Definition . *Average of a set of N numbers is always equal to the average of sub set average of N numbers.*

Let $N = \{n_1, n_2, n_3, \dots, n_k\}$ be a set of k numbers with average AVGN .We can form i subsets from N where $k \geq i \geq 1$; If $AVGS_i$ is the average of i^{th} subset then $AVGN = 1/i \sum_{i=1}^k AVGS_i$ is true.

Table 3. Numeric data perturbation where original numbers are replaced by their bin averages

F1	F2	Root Average	Division around root avg	Lvel1 Aver ages	Division Around Level avg	Final Aver ages	R E P	Original Values	Pertu rbed values
aa	12	} 26.4	34	43.6	54	54	L	12	9.5
bbb	20		43		43	38.5	A	20	17.5
cc	34		54		34		C	34	38.5
ddd	54		7	13.5	20	17.5	I	54	54
xxx	43		12		15		N	43	38.5
yyy	15		15		12	9.5	G	15	17.5
zzz	7		20		7		7	7	9.5

Table-3 illustrates our scheme for numeric data perturbation. Here the complete data set is divided and conquered around the arithmetic average recursively, till the bin size reaches specified minimum. In our approach bin size is not fixed but computed dynamically depending upon the minimum value of level averages.

Average of perturbed values=26.5 (suggests summary statistic do not change even after perturbation by binning by averages)

4 Algorithms

Algorithm-1 : Binning by averages

- Step 0: Let j be the attribute that is to be perturbed by binning.
- Step 1: Define the stopping criterion, minimum size of the subset(bin)
- Step 2: Find the average of the attribute j .
- Step 3: Partition j into two subsets based on the average.
- Step 4: Repeat steps 2 and 3 on each of the subsets.
- Step 5: Check the minimum size of the bin, stop if stopping criterion is met.
- Step 5: Replace each value in j by its subset average.

Algorithm-2: Rotation number computation and field rotation

- Step 1 : Mine the table and get the association rules.
- Step 2 : Compute the support for each association rule.
- Step 3 : Get the minimum support value and identify the valid association rules.
- Step 4: Fix k value ($1 \leq k \leq 10$) and compute $S_k = (k * S_a)$ for each valid association rule, S_a is the actual support of the association rule.
- Step 5: In $X \Rightarrow Y$ kind of association rules, For each rotation in X , rotate all Y member attributes and compute the support values in each case of rotation and record the number of corresponding rotations.
- Step 6: Identify the rotation on X and Y that gives S_k value of support.
- Step 7: Repeat step 5 and 6 for each association rule obtained in step 3 and Sum the number of rotations if similar attribute is encountered in $X \Rightarrow Y$ rule.
- Step 8: To obtain the final indicator about the number of rotation on each attribute Get the summed value on number of rotations from step 7 and operate modulus R on this number, R is the total number of tuples in the given table. The resulting integer is the exact rotations to be carried out on that attribute.
- Step 9: Lock other columns in the table and rotate the only column containing the attributes visible in step 8.
- Step 10: Repeat step 9 for all valid attributes and get the perturbed table.

5 Experiments and Results

We tested our implementations on market basket dataset and weather condition data set. Market basket data is from the source [16]. This data set contained 18 attributes and 1000 tuples. The table contains attributes like personal information and purchased items information. Weather condition data is used for weather forecast. Weather condition data is obtained from the source [16]. This data set contains 7 attributes and 4184 tuples. The weather condition table contains attributes like pressure, temperature, time, power etc.

The number of association rules generated before perturbation and after perturbation for a minimum support of 40% is found as recorded in table-4.

Table 4. Number of Association rules generated for min_sport=40%

Dataset	Number of Association rules generated		Percentage of matching in rules
	Before perturbation	After perturbation	
Market basket data	66	62	92%
Weather condition	38	34	93%

We trained a Neural Net classifier from [16] with unperturbed data sets and obtained estimated and analysis accuracies table-5 for the selected data sets. Then we perturbed each dataset by rotating the fields

Table 5. Classifier accuracies before and after perturbation

Market basket dataset		
Perturbation state	Estimated Accuracy	Analysis Accuracy
NO	57.94	59.12
YES	57.28	59.20
Weather Condition dataset		
Perturbation state	Estimated Accuracy	Analysis Accuracy
NO	98.36	98.71
YES	98.62	98.72

6 Conclusion and Future Work

Disclosing distorted data instead of the original data is a natural way to protect privacy. The essential requirement by any such data distortion method is that the summary statistics of the data should not change in the process of distortion, otherwise the data becomes irrelevant for analysis. Therefore revealing information while preserving privacy is a challenge. We presented a method where in summary statistics such

as average of the data is not changing even after perturbing by bin averages. The field rotation is a simple method and when the number of rotations to be carried out on each field is made as a dependent function on the support value of the association rule, it helps to preserve summary statistics from association rule mining. Future work lies in giving a theoretical proof for the simple observations made in this paper and in conducting some more tests with standard data sets. Complexity analysis of the algorithms developed in the paper and adoption of the perturbation techniques in real world applications need to be looked into to increase the practical importance of the observations.

References

1. Estivill-Castro, V., Brankovic, L.: Data Swapping: Balancing Privacy against Mining of Association Rules. In: Proceedings of Knowledge Discovery and Data Warehousing, Florence, Italy, August 1999, pp. 389–398 (1999)
2. Muralidhar, K., Parsa, R., Sarathy, R.: A general additive data perturbation method for database security. *Management Science* 45(10), 1399–1415 (1999)
3. Agrawal, R., Srikant, R.: Privacy-preserving data mining. In: Proceedings of the 2000 ACM SIGMOD International Conference on dataset Management of Data, Dallas, Texas (May 2000)
4. Pierangela, S.: Protecting Respondents' Identities in Microdata Release. *IEEE Transactions on Knowledge and Data engineering*, 13(6) (November-December 2001)
5. Dasseni, E., Verykois, V.S., Elmagarid, A.K., Bertino, E.: Hiding Association rules by using Confidence and Support. In: Proceedings of Information Hiding Workshop, pp. 369–383 (2001)
6. Sweeny, L.: K-anonymity a model for protecting privacy. *International journal on uncertainty, Fuzzyness and knowledge based systems*, (5), 557–570 (2002)
7. Domingo-Ferrer, J., Mateo-Sanz, J.M.: Practical Data-Oriented Micro-aggregation for Statistical Disclosure Control. *IEEE Transaction on Knowledge and Data Eng.* 14(1), 189–201 (2002)
8. Datta, S., Kargupta, H., Sivakumar, K.: Homeland defense, privacy sensitive data mining, and random value distortion. In: Proceedings of the SIAM Workshop on Data Mining for Counter Terrorism and Security (SDM 2003), San Fransisco, C.A (May 2003)
9. Kargupta, H., Datta, S., Wang, Q., Sivakumar, K.: On the privacy preserving properties of random data perturbation techniques. In: Proceedings of the third IEEE International Conference on Data mining (ICDM 2003), Melbourne, Florida, November 19-22 (2003)
10. Verykios, V., Bertino, E., Nai, I., Loredana, F., Provenza, P., Saygin, Y., Theodoriddis, Y.: State of the Art in Privacy preserving Data Mining. In: SIGMOD Record, vol. 33(1) (2004)
11. Bakken, D., Parameswaran, R., Blough, D.: Data Obfuscation: Anonymity and Desensitization of Usable data Sets. In: *IEEE Security and Privacy*, vol. 2, pp. 34–41 (November-December 2004)
12. Hillol, K., Souptik, D., Oi, W., Krishnamurthy, S.: Random-data perturbation techniques and privacy preserving data mining. In: *Knowledge and Information Systems*, May 2005, vol. 7(4) (2005)
13. Chawla, S., Dwork, F., McSherry, Smith, A., Wee, H.: Towards privacy in public databases. In: *Theory of cryptography conference*, Cambridge, MA, February 9-12 (2005)

14. Li, L., Murat, K., Bhavani, T.: The Applicability of the Perturbation Model-based privacy Preserving Data Mining for real-world Data. In: Sixth IEEE International Conference on Data Mining – Workshops ICDMW 2006 (2006)
15. Vaidya, J., Christopher, W., Yu, C., Zhu, M.: Privacy Preserving Data Mining, vol. 13. Springer, Heidelberg (2006)
16. Clementine Workbench, <http://www.spss.com>
17. Ciriani, V., De Capitani, S., di Vimercati, S., Samarati, F.P.: K-anonymity. Secure Data Management. In: Decentralized Systems (2007)
18. Wu, Y.-H., Chiang, C.-M., Arbee, L., Chen, P.: Hiding Sensitive Association Rules with Limited Side Effects. IEEE Transaction on Knowledge and Data Engineering, vol. 19(1) (January 2007)
19. Fung, B., Wang, C.M., Ke, Y., Philip, S.: Anonymizing Classification Data for Privacy preservation. In: IEEE Transactions on Knowledge and Data Engineering, May 2007, vol. 19(5), pp. 711–725 (2007)
20. Xu, S., Lai, S.: Fast Fourier Transform Based Data Perturbation Method for Privacy Protection. IEEE Transactions on Intelligence and Security Informatics, 221–224 (May 2007)

Mining Weighted Frequent Patterns Using Adaptive Weights

Chowdhury Farhan Ahmed, Syed Khairuzzaman Tanbeer,
Byeong-Soo Jeong, and Young-Koo Lee

Department of Computer Engineering, Kyung Hee University
1 Seochun-dong, Kihung-gu, Youngin-si, Kyunggi-do, 446-701, Republic of Korea
 {farhan,tanbeer,jeong,yklee}@khu.ac.kr

Abstract. Existing weighted frequent pattern (WFP) mining algorithms assume that each item has fixed weight. But in our real world scenarios the weight (price or significance) of an item can vary with time. Reflecting such change of weight of an item is very necessary in several mining applications such as retail market data analysis and web click stream analysis. In this paper, we introduce a novel concept of adaptive weight for each item and propose an algorithm AWFPM (adaptive weighted frequent pattern mining). Our algorithm can handle the situation where the weight (price or significance) of an item may vary with time. Extensive performance analyses show that our algorithm is very efficient and scalable for WFP mining using adaptive weights.

Keywords: Data mining, knowledge discovery, weighted frequent pattern mining, adaptive weight.

1 Introduction

Weighted frequent pattern (WFP) mining is more practical than traditional frequent pattern mining [1], [7], [10], [11] because it can consider different semantic significance (weight) of items. In many cases, the item in a transaction can have different degree of importance (weight). For example, in retail applications the expensive product may contribute a large portion of overall revenue even though it does not appear in a large number of transactions. For this reason, WFP mining [2], [3], [4], [5], [6] was proposed to discover more important knowledge considering different weights of each item, which plays an important role in the real world scenarios. Weight based pattern mining approach can be applied in many areas, such as market data analysis where the prices of products are important factors, web traversal pattern mining where each web page has different strength of impact, and biomedical data analysis where most diseases are not caused by a single gene but by a combination of genes.

Even though WFP mining can consider application-specific diverse weights of each item during the mining process, it is not enough to reflect the real world environment where the significance (weight) of an item can vary with time. In real world scenarios, the significance of each item might be widely affected by many

factors. Peoples' buying behaviors (or interests) are changing with time, so they may affect the significances (weights) of products in retail markets. The weights of seasonal products may also vary when the season changes from summer to winter or winter to summer. Web click stream analysis can be another example of this. The significance of each web page (or web site) may change with time depending on the popularity, political issues, public events, and so on.

Motivated by these real world scenarios, we propose a new strategy for handling adaptive weights in WFP mining. As in the real world business market the weight (importance or price) of an item may vary due to the environmental change in different time periods, in this paper, we introduce a novel concept of adaptive weight for each item in WFP mining. Our proposed approach keeps track of varying weights of each item batch by batch in a prefix tree. To handle the adaptive weights during the mining process we propose a new algorithm AWFPM (adaptive weighted frequent pattern mining). Our algorithm can handle the situation where the weight (importance or price) of an item may vary with time. It exploits a pattern growth mining technique to avoid the level-wise candidate generation-and-test problem. Furthermore, it requires only one scan of a database and therefore eligible for stream data mining [8], [9].

The remainder of this paper is organized as follows. In Section 2, we describe background. In Section 3, we explain our proposed AWFPM algorithm for weighted frequent pattern mining using adaptive weights. In Section 4, our experimental results are presented and analyzed. Finally, in Section 5, conclusions are presented.

2 Background

A weight of an item is a non-negative real number which is assigned to reflect the importance of each item in the transaction database [2], [3]. For a set of items $I = \{i_1, i_2, \dots, i_n\}$, weight of a pattern $P\{x_1, x_2, \dots, x_m\}$ is given as follows:

$$Weight(P) = \frac{\sum_{q=1}^{length(P)} Weight(x_q)}{length(P)} \quad (1)$$

A weighted support of a pattern is defined as the resultant value of multiplying the pattern's support with the weight of the pattern [2], [3]. So the weighted support of a pattern P is given as follows:

$$Wsupport(P) = Weight(P) \times Support(P) \quad (2)$$

A pattern is called a weighted frequent pattern if the weighted support of the pattern is greater than or equal to the minimum threshold [2], [3].

In the very beginning some weighted frequent pattern mining algorithms WARM [5], WAR [6] have been developed based on the Apriori algorithm [1] using candidate generation-and-test paradigm. Obviously, these algorithms require multiple database scans and result in poor mining performance. WFIM [2] is the first FP-tree [7] based weighted frequent pattern algorithm using two

database scans over a static database. They have used a minimum weight and a weight range. Items are given fixed weights randomly from the weight range. They have arranged the FP-tree [7] in weight ascending order. To extract the more interesting weighted frequent patterns, WIP [3] algorithm introduces a new measure of weight-confidence to measure strong weight affinity of a pattern.

WFIM [2] and WIP [3] showed that the main challenging problem of weighted frequent pattern mining is that the weighted frequency of an itemset (or a pattern) does not have the *downward closure* property as in frequent pattern mining. This property tells that if a pattern is infrequent then all of its super patterns must be infrequent. Consider the item “a” has weight 0.6 and frequency 4, the item “b” has weight 0.2 and frequency 5, the itemset “ab” has frequency 3. According to equation (1), the weight of the itemset “ab” will be $(0.6 + 0.2)/2 = 0.4$ and according to equation (2) its weighted frequency will be $0.4 \times 3 = 1.2$. Weighted frequency of “a” is $0.6 \times 4 = 2.4$ and “b” is $0.2 \times 5 = 1.0$. If the minimum threshold is 1.2, then pattern “b” is weighted infrequent but “ab” is weighted frequent. WFIM and WIP maintain the *downward closure* property by multiplying each itemset’s frequency by the global maximum weight. In the above example, if item “a” has the maximum weight which is 0.6, then by multiplying it with the frequency of item “b”, 3.0 can be obtained. So, pattern “b” is not pruned at the early stage and pattern “ab” will not be missed.

Zhang et al. [4] proposed a new strategy (“Weight”) for maintaining the association rules in incremental databases by using the weighting technique to highlight new data. Any recently added transactions are assigned higher weights. Moreover, all transactions in a database are given the same weight. They did not use different weights for individual items or transactions. Their algorithm is based on the level-wise candidate set generation-and-test methodology of the Apriori [1] algorithm. Therefore, for a particular dataset, they generate a large number of candidates and need to perform several database scans to get the final result.

In the existing weighted frequent pattern mining works, no one proposed any solution for adaptive weighted frequent pattern mining, where the weight of an item may be changed in any batch of transactions. Therefore, we propose a novel algorithm for adaptive weighted frequent pattern mining.

3 Our Proposed Algorithm

Definition 1. Adaptive weighted support of a pattern P is defined by

$$AWsupport(P) = \sum_{j=1}^N Weight_j(P) \times Support_j(P) \tag{3}$$

Here N is the number of batches. $Weight_j(P)$ and $Support_j(P)$ are the weight and support of pattern P respectively in the j^{th} batch. We can calculate the value of $Weight_j(P)$ by using equation (1). For example, the $AWsupport$ of pattern “bd” in the first, second and third batches are $((0.9 + 0.3) / 2) \times 1 = 0.6$, $((0.7$

Table 1. An example of transaction database with adaptive weights

Batch	TID	Trans.	Weight				
1 st	T ₁	a, b, d	a	b	c	d	e
	T ₂	c, d	0.45	0.9	0.2	0.3	0.5
	T ₃	a, b					
2 nd	T ₄	b	a	b	c	d	e
	T ₅	b, c, d	0.6	0.7	0.4	0.5	0.4
	T ₆	c, e					
3 rd	T ₇	a, c, e	a	b	c	d	e
	T ₈	a	0.5	0.3	0.7	0.4	0.45
	T ₉	a, c					

HeaderTable

Item	W	F
a	0.45, 0.6, 0.5	2, 0, 3
b	0.9, 0.7, 0.3	2, 2, 0
c	0.2, 0.4, 0.7	1, 2, 2
d	0.3, 0.5, 0.4	2, 1, 0
e	0.5, 0.4, 0.45	0, 1, 1

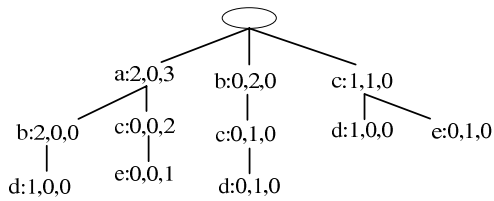


Fig. 1. Constructed tree after inserting 1st, 2nd and 3rd Batches

$+ 0.5) / 2) \times 1 = 0.6$ and $((0.3 + 0.4) / 2) \times 0 = 0$ respectively in Table 1. So, total $AWsupport("bd") = 0.6 + 0.6 + 0 = 1.2$.

Definition 2. A pattern is called a adaptive weighted frequent pattern if the adaptive weighted support of the pattern is greater than or equal to the minimum threshold. For example, if the minimum threshold is 1.2, then “bd” is a adaptive weighted frequent pattern in Table 1.

At first we describe the construction process of our tree structure to capture transactions having items with adaptive weights. Header table is also maintained in our tree like FP-tree [7]. The first value of the header table is the item id. After that the frequency and weight information of an item is kept in batch by batch fashion inside the header table. The tree nodes only contain item id and its batch by batch frequency information. To facilitate the tree traversals adjacent links are also maintained (not shown in the figure for simplicity) in our tree structure like FP-tree.

Consider the example database shown in Table 1. After reading a transaction from database, at first the items inside it are sorted according to lexicographical order and then they are inserted into the tree. Fig. 1 shows the tree after capturing the transactions of batch 1, 2 and 3. As batch-by-batch frequency information is kept separately in each node of the tree, we can easily discover that which transactions have been occurred in which batch. For example, we can easily detect that the batch number of transaction “c, d” and “c, e” is 1

and 2 respectively from Fig. 1. Our tree structure holds the following important property.

Property 1. The total count of frequency values of any node in the tree is greater than or equal to the sum of total counts of frequency values of its children.

Now we discuss the mining process of our proposed AWFPM algorithm using FP-growth approach [7]. As discussed in Section 2, the main challenging problem in this area is that the weighted frequency of an itemset does not have the *downward closure* property and to maintain this property we have to use the global maximum weight. The global maximum weight is the maximum weight of all the items in the global database. In our case, global maximum weight is the highest weight among all the weights in all batches. For example, in Table 1, item “*b*” has the global maximum weight 0.9. We will refer this term as *GMAXW*. Local maximum weight is needed when we are doing the mining operation for a particular item. It is not always the *GMAXW*. For example, in the database of Table 1, item “*e*” has not been occurred with item “*b*” and “*d*”. As a result, in the mining operation the conditional tree of “*e*” can only contain items “*a*” and “*c*”. Here we do not need to use *GMAXW*. Local maximum weight can maintain the *downward closure* property. The local maximum weight for “*e*” is the maximum weight of the items “*c*”, “*a*” and “*e*”, which is 0.7. We will refer local maximum weight as *LMAXW*. Using *LMAXW* in place of *GMAXW* reduces the probability of a pattern to be a candidate.

Consider the database presented at Table 1, tree constructed for that database in Fig. 1 and minimum threshold = 1.2. Here *GMAXW* = 0.9. After multiplying the total frequency of each item with *GMAXW*, the adaptive weighted frequency list is $\langle a : 4.5, b : 3.6, c : 4.5, d : 2.7, e : 1.8 \rangle$. As a result, all items are single element candidates. Now we construct the conditional trees for these items in a bottom up fashion and mine the adaptive weighted frequent patterns. At first conditional tree of bottom most item “*e*” is created by taking all the branches prefixing item “*e*” and deleting the nodes containing an item which can not be a candidate pattern with item “*e*” shown in Fig. 2(a). For item “*e*”, *LMAXW* = 0.7 as it has occurred only item “*a*” and “*c*”. Item “*a*” and “*c*” has total frequency 1 and 2 respectively with item “*e*”. So, after multiplying these frequencies with *LMAXW* = 0.7, we get the adaptive weighted frequency list $\langle a : 0.7, c : 1.4 \rangle$ for item “*e*”. As item “*a*” has low adaptive weighted frequency with item “*e*”, it has to be deleted to get the conditional tree of “*e*”. As a result, Candidate patterns “*ce*” and “*e*” are generated here.

For the item “*d*”, the *LMAXW* = 0.9 and adaptive weighted frequency list is $\langle a : 0.9, b : 1.8, c : 1.8 \rangle$. By deleting item “*a*” we get the conditional tree of item “*d*” (shown in Fig. 2(b)). Candidate patterns “*bd*”, “*cd*” and “*d*” are generated here. For the pattern “*dc*”, the adaptive weighted frequency list is $\langle b : 0.9 \rangle$. As a result, no conditional tree or candidate pattern is generated for pattern “*dc*”. For the item “*c*”, the *LMAXW* = 0.9 and adaptive weighted frequency list is $\langle a : 1.8, b : 0.9 \rangle$. By deleting item “*b*” we get the conditional tree of item “*c*” (shown in Fig. 2(c)). Candidate patterns “*ac*” and “*c*” are generated here. For the item “*b*”, the *LMAXW* = 0.9 and adaptive weighted

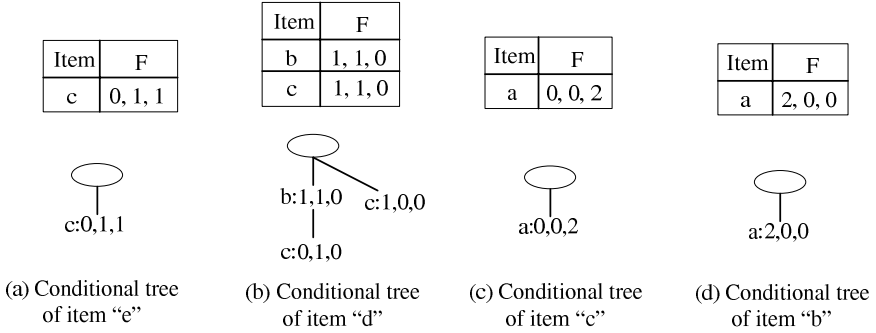


Fig. 2. Mining Operation

frequency list is $\langle a : 1.8 \rangle$. Conditional tree of item “b” is created in Fig. 2(d) and patterns “ab” and “b” are generated here. Last candidate pattern “a” is generated for the top most item “a”. After testing all these candidate patterns with their actual adaptive weighted frequency using equation (3), we can get six adaptive weighted frequent patterns. Here are the six adaptive weighted frequent patterns with their adaptive weighted support: $\langle b, d : 1.2 \rangle$, $\langle a, c : 1.2 \rangle$, $\langle c : 2.4 \rangle$, $\langle a, b : 1.35 \rangle$, $\langle b : 3.2 \rangle$ and $\langle a : 2.4 \rangle$.

4 Experimental Results

To evaluate the performance of our proposed algorithm, we have performed several experiments on both IBM synthetic dataset (T10I4D100K) and real-life dataset mushroom from frequent itemset mining dataset repository (<http://fimi.cs.helsinki.fi/data/>). These datasets do not provide the weight values of each item. As like the performance evaluation of the previous weight based frequent pattern mining [2, 3, 4, 5, 6], we have generated random numbers for the weight values of each item, ranging from 0.1 to 0.9. We compare the performance of our algorithm with the existing algorithm, “Weight” [4]. Our programs were written in Microsoft Visual C++ 6.0 and run with Windows XP operating system on a Pentium dual core 2.13 GHz CPU with 1GB main memory.

The mushroom dataset contains 8,124 transactions and 119 distinct items. Its mean transaction size is 23, and it is a dense dataset. Almost 20% ($(23/119) \times 100$) of its distinct items are present in every transaction. We have divided this dataset into 3, 6 and 9 batches. From now we will denote N to represent the number of batches. When $N = 3$, the first and second batches contain 3000 transactions and the third/last batch contains 2124 transactions. For $N = 6$ and 9, all the batches contain 1500 and 1000 transactions respectively except the last batch. The numbers of transactions in the last batches are the remaining transactions at the last. For each batch we have generated new weight value of each item. We have used $N = 3$ for the existing algorithm, “Weight”. Fig. 3

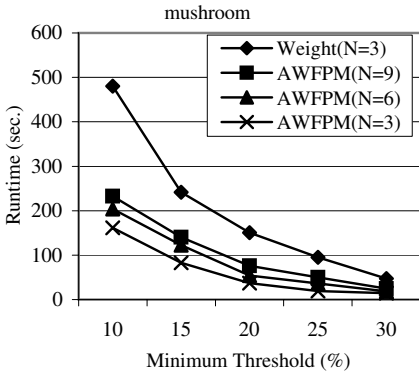


Fig. 3. Performance on mushroom

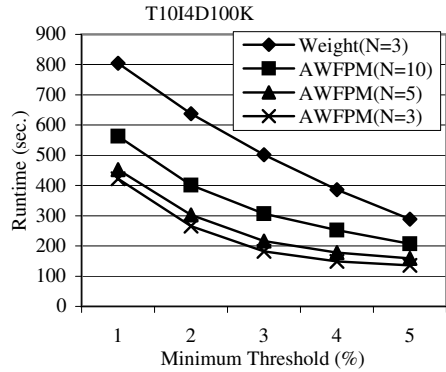


Fig. 4. Performance on T10I4D100K

shows the runtime performance curves for mushroom. The minimum threshold range of 10% to 30% is used in Fig. 3.

The T10I4D100K dataset contains 100,000 transactions and 870 distinct items. Its mean transaction size is 10.1, and it is a sparse dataset. Around 1.16% $((10.1 / 870) \times 100)$ of its distinct items are present in every transaction. We have divided this dataset into 3, 5 and 10 batches. For $N = 5$ and 10, all the batches contain 20000 and 10000 transactions respectively. For $N = 3$, the first two batches contain 35000 transactions and the last batch contains 30000 transactions. For each batch we have generated new weight value of each item. We have used $N = 3$ for the existing algorithm “Weight”. Fig. 4 shows the runtime performance curves for T10I4D100K. The minimum threshold range of 1% to 5% is used in Fig. 4. The experimental results in Figs. 3 and 4 demonstrate that by using an efficient tree structure, the single database scan approach, and the pattern growth mining technique, our algorithm performs better than the existing algorithm.

Prefix tree based frequent pattern mining research work [10], [11] showed that the memory requirements for the prefix trees are quite possible and efficient by using the gigabyte-range memory available recently. Our proposed AWFPM algorithm can also mine adaptive weighted frequent patterns efficiently by using a prefix tree. Moreover, it can save huge memory space by keeping batch by batch transaction information in the prefix tree in a compact format. It requires much less memory compared to the existing algorithm “Weight”. In the mushroom dataset ($N=3$), our algorithm and the existing algorithm require 0.59 MB and 1.26 MB memory respectively. In the T10I4D100K dataset ($N=3$), our algorithm and the existing algorithm require 12.72 MB and 17.61 MB memory respectively.

5 Conclusions

In this paper we introduced a novel concept of adaptive weight for each item in weighted frequent pattern mining and we provided an algorithm to efficiently mine adaptive weighted frequent patterns. Our main goal is to find the weighted

frequent patterns using adaptive weights that can truly reflect the real world scenarios. By keeping batch by batch frequency and weight information, our proposed algorithm AWFPM can accurately mine the adaptive weighted frequent patterns. Moreover, it exploits a pattern growth mining technique to avoid the level-wise candidate generation-and-test problem. Our approach is also applicable in real time data processing like data stream as it requires only one database scan. Extensive performance analyses show that our algorithm is efficient in both dense and sparse datasets to mine adaptive weighted frequent patterns.

Acknowledgments. This study was supported by a grant of the Korea Health 21 R&D Project, Ministry For Health, Welfare and Family Affairs, Republic of Korea(A020602).

References

1. Agrawal, R., Srikant, R.: Fast Algorithms for Mining Association Rules. In: 20th Int. Conf. on Very Large Data Bases (VLDB), pp. 487–499 (1994)
2. Yun, U., Leggett, J.J.: WFIM: weighted frequent itemset mining with a weight range and a minimum weight. In: Fourth SIAM Int. Conf. on Data Mining, USA, pp. 636–640 (2005)
3. Yun, U.: Efficient Mining of weighted interesting patterns with a strong weight and/or support affinity. *Information Sciences* 177, 3477–3499 (2007)
4. Zhang, S., Zhang, C., Yan, X.: Post-mining: maintenance of association rules by weighting. *Information Systems* 28, 691–707 (2003)
5. Tao, F.: Weighted association rule mining using weighted support and significant framework. In: 9th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining, USA, pp. 661–666 (2003)
6. Wang, W., Yang, J., Yu, P.S.: WAR: weighted association rules for item intensities. *Knowledge Information and Systems* 6, 203–229 (2004)
7. Han, J., Pei, J., Yin, Y., Mao, R.: Mining frequent patterns without candidate generation: a frequent-pattern tree approach. *Data Mining and Knowledge Discovery* 8, 53–87 (2004)
8. Jiang, N., Gruenwald, L.: Research Issues in Data Stream Association Rule Mining. *SIGMOD Record* 35(1), 14–19 (2006)
9. Leung, C.K.-S., Khan, Q.I.: DSTree: A Tree structure for the mining of frequent Sets from Data Streams. In: Perner, P. (ed.) *ICDM 2006*. LNCS (LNAI), vol. 4065, pp. 928–932. Springer, Heidelberg (2006)
10. Leung, C.K.-S., Khan, Q.I., Li, Z., Hoque, T.: CanTree: a canonical-order tree for incremental frequent-pattern mining. *Knowledge and Information Systems* 11(3), 287–311 (2007)
11. Tanbeer, S.K., Ahmed, C.F., Jeong, B.: CP-tree: A tree structure for single pass frequent pattern mining. In: Washio, T., Suzuki, E., Ting, K.M., Inokuchi, A. (eds.) *PAKDD 2008*. LNCS (LNAI), vol. 5012, pp. 1022–1027. Springer, Heidelberg (2008)

On the Improvement of the Mapping Trustworthiness and Continuity of a Manifold Learning Model

Raúl Cruz-Barbosa^{1,2} and Alfredo Vellido¹

¹ Universitat Politècnica de Catalunya, 08034, Barcelona, Spain
{rcruz,avellido}@lsi.upc.edu

² Universidad Tecnológica de la Mixteca, 69000, Huajuapán, Oaxaca, México

Abstract. Manifold learning methods model high-dimensional data through low-dimensional manifolds embedded in the observed data space. This simplification implies that they are prone to trustworthiness and continuity errors. Generative Topographic Mapping (GTM) is one such manifold learning method for multivariate data clustering and visualization, defined within a probabilistic framework. In the original formulation, GTM is optimized by minimization of an error that is a function of Euclidean distances, making it vulnerable to the aforementioned errors, especially for datasets of convoluted geometry. Here, we modify GTM to penalize divergences between the Euclidean distances from the data points to the model prototypes and the corresponding geodesic distances along the manifold. Several experiments with artificial data show that this strategy improves the continuity and trustworthiness of the data representation generated by the model.

1 Introduction

Manifold learning methods model high-dimensional data under the assumption that they can be faithfully represented by a low-dimensional manifold embedded in the observed data space. This simplifying assumption makes these methods prone to two types of potential errors: data point neighbourhood relations that are not preserved in their low-dimensional representation, which hamper the continuity of the latter, and spurious neighbouring relations in the representation that do not have a correspondence in the observed space, which limit the trustworthiness of the low-dimensional representation.

Amongst density-based methods, Finite Mixture Models have established themselves as flexible and robust tools for multivariate data clustering [1]. In order to endow Finite Mixture Models with data visualization capabilities, which are important for data exploration, certain constraints must be enforced. One alternative is forcing the mixture components to be centred in a low-dimensional manifold embedded into the usually high-dimensional observed data space. Such approach is the basis for the definition of Generative Topographic Mapping (GTM) [2], a flexible manifold learning model for simultaneous data clustering and visualization whose probabilistic nature makes possible to extend it to

perform tasks such as missing data imputation [3], robust handling of outliers, and unsupervised feature selection [4], amongst others.

In the original formulation, GTM is optimized by minimization of an error that is a function of Euclidean distances, making it vulnerable to continuity and trustworthiness problems, especially for datasets of complex and convoluted geometry. Such data may require plenty of folding from the GTM model, resulting in an unduly entangled embedded manifold that would hamper both the visualization of the data and the definition of clusters the model is meant to provide. Following an idea proposed in [5], the learning procedure of GTM is here modified by penalizing the divergences between the Euclidean distances from the data to the model prototypes and the corresponding approximated geodesic distances along the manifold. In this paper, we assess to what extent the resulting Geo-GTM model, which incorporates the data visualization capabilities that the model proposed in [5] lacks, is capable of preserving the trustworthiness and continuity of the mapping.

2 Manifolds and Geodesic Distances

Manifold methods such as ISOMAP [6] and Curvilinear Distance Analysis [7] use the geodesic distance as a basis for generating the data manifold. This metric favours similarity along the manifold, which may help to avoid some of the distortions that the use of a standard metric such as the Euclidean distance may introduce when learning the manifold. In doing so, it can avoid the breaches of topology preservation that may occur due to excessive folding.

The otherwise computationally intractable geodesic metric can be approximated by graph distances [8], so that instead of finding the minimum arc-length between two data points lying on a manifold, we would set to find the shortest path between them, where such path is built by connecting the closest successive data points. In this paper, this is done using the K -rule, which allows connecting the K -nearest neighbours. A weighted graph is then constructed by using the data and the set of allowed connections. The data are the vertices, the allowed connections are the edges, and the edge labels are the Euclidean distances between the corresponding vertices. If the resulting graph is disconnected, some edges are added using a minimum spanning tree procedure in order to fully connect it. Finally, the distance matrix of the weighted undirected graph is obtained by repeatedly applying Dijkstra's algorithm [9], which computes the shortest path between all data samples.

3 GTM and Geo-GTM

The standard GTM is a non-linear latent variable model defined as a mapping from a low dimensional latent space onto the multivariate data space. The mapping is carried through by a set of basis functions generating a constrained mixture density distribution. It is defined as a generalized linear regression model:

$$\mathbf{y} = \phi(\mathbf{u})\mathbf{W}, \quad (1)$$

where ϕ are R basis functions, Gaussians in the standard formulation; \mathbf{W} is a matrix of adaptive weights w_{rd} ; and \mathbf{u} is a point in latent space. To avoid computational intractability, a regular grid of M points \mathbf{u}_m can be sampled from the latent space, which acts as visualization space. Each of them, which can be considered as the representative of a data cluster, has a fixed prior probability $p(\mathbf{u}_m) = 1/M$ and is mapped, using (II), into a low-dimensional manifold non-linearly embedded in the data space. A probability distribution for the multivariate data $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^N$ can then be defined, leading to the following expression for the log-likelihood:

$$L(\mathbf{W}, \beta | \mathbf{X}) = \sum_{n=1}^N \ln \left\{ \frac{1}{M} \sum_{m=1}^M \left(\frac{\beta}{2\pi} \right)^{D/2} \exp \left\{ -\beta/2 \|\mathbf{y}_m - \mathbf{x}_n\|^2 \right\} \right\} \quad (2)$$

where \mathbf{y}_m , usually known as *reference* or *prototype* vectors, are obtained for each \mathbf{u}_m using (II); and β is the inverse of the noise variance, which accounts for the fact that data points might not strictly lie on the low dimensional embedded manifold generated by the GTM. The EM algorithm is an straightforward alternative to obtain the Maximum Likelihood (ML) estimates of the adaptive parameters of the model, namely \mathbf{W} and β . In the E-step, the responsibilities z_{mn} (the posterior probability of cluster m membership for each data point \mathbf{x}_n) are computed as

$$z_{mn} = p(\mathbf{u}_m | \mathbf{x}_n, \mathbf{W}, \beta) = \frac{p(\mathbf{x}_n | \mathbf{u}_m, \mathbf{W}, \beta) p(\mathbf{u}_m)}{\sum_{m'} p(\mathbf{x}_n | \mathbf{u}_{m'}, \mathbf{W}, \beta) p(\mathbf{u}_{m'})}, \quad (3)$$

where $p(\mathbf{x}_n | \mathbf{u}_m, \mathbf{W}, \beta) = \mathcal{N}(\mathbf{y}(\mathbf{u}_m, \mathbf{W}), \beta)$.

3.1 Geo-GTM

The Geo-GTM model is an extension of GTM that favours the similarity of points along the learned manifold, while penalizing the similarity of points that are not contiguous in the manifold, even if close in terms of the Euclidean distance. This is achieved by modifying the standard calculation of the responsibilities in (3) proportionally to the discrepancy between the geodesic (approximated by the graph) and the Euclidean distances. Such discrepancy is made operational through the definition of the exponential distribution

$$\mathcal{E}(d_g | d_e, \alpha) = \frac{1}{\alpha} \exp \left\{ -\frac{d_g(\mathbf{x}_n, \mathbf{y}_m) - d_e(\mathbf{x}_n, \mathbf{y}_m)}{\alpha} \right\}, \quad (4)$$

where $d_e(\mathbf{x}_n, \mathbf{y}_m)$ and $d_g(\mathbf{x}_n, \mathbf{y}_m)$ are, in turn, the Euclidean and graph distances between data point \mathbf{x}_n and the GTM prototype \mathbf{y}_m . Responsibilities are redefined as:

$$z_{mn}^{geo} = p(\mathbf{u}_m | \mathbf{x}_n, \mathbf{W}, \beta) = \frac{p'(\mathbf{x}_n | \mathbf{u}_m, \mathbf{W}, \beta) p(\mathbf{u}_m)}{\sum_{m'} p'(\mathbf{x}_n | \mathbf{u}_{m'}, \mathbf{W}, \beta) p(\mathbf{u}_{m'})}, \quad (5)$$

where $p'(\mathbf{x}_n | \mathbf{u}_m, \mathbf{W}, \beta) = \mathcal{N}(\mathbf{y}(\mathbf{u}_m, \mathbf{W}), \beta) \mathcal{E}(d_g(\mathbf{x}_n, \mathbf{y}_m)^2 | d_e(\mathbf{x}_n, \mathbf{y}_m)^2, 1)$. When there is no agreement between the graph approximation of the geodesic distance

and the Euclidean distance, the value of the numerator of the fraction within the exponential in (4) increases, pushing the exponential and, as a result, the modified responsibility, towards smaller values, i.e., punishing the discrepancy between metrics. Once the responsibility is calculated in the modified E-step, the rest of the model’s parameters are estimated following the standard EM procedure. Each data point \mathbf{x}_n can then be visualized in latent space as the mean of the estimated posterior distribution:

$$\mathbf{u}_n^{mean} = \sum_{m=1}^M \mathbf{u}_m z_{mn}^{geo}, \tag{6}$$

4 Experiments

Geo-GTM was implemented in MATLAB®. For the experiments reported next, the adaptive matrix \mathbf{W} was initialized, following a procedure described in [2], as to minimize the difference between the prototype vectors \mathbf{y}_m and the vectors that would be generated in data space by a partial Principal Component Analysis (PCA). The inverse variance β was initialised to be the inverse of the 3^{rd} PCA eigenvalue. This initialization ensures the replicability of the results. The latent grid was fixed to a square layout of approximately $(N/2)^{1/2} \times (N/2)^{1/2}$, where N is the number of points in the dataset. The corresponding grid of basis functions was equally fixed to a 5×5 square layout for all datasets.

4.1 Quantitative Evaluation of the Mappings

As mentioned in the introduction, data neighbourhood relations that are not preserved in the low-dimensional representation, hamper the continuity of the latter, while spurious neighbouring relations in the low-dimensional representation that do not have a correspondence in the observed space limit its trustworthiness. In order to evaluate and compare the mappings generated by GTM and Geo-GTM, we use here the trustworthiness and continuity measures developed in [10]. Trustworthiness is defined as:

$$T(K) = 1 - \frac{2}{NK(2N - 3K - 1)} \sum_{i=1}^N \sum_{x_j \in U_K(x_i)} (r(x_i, x_j) - K), \tag{7}$$

where $U_k(x_i)$ is the set of data points x_j for which $x_j \in \hat{C}_K(x_i) \wedge x_j \notin C_K(x_i)$ and $C_K(x_i)$ and $\hat{C}_K(x_i)$ are the sets of K data points that are closest to x_i in the observed data space and in the low-dimensional representation space, respectively. Continuity is in turn defined as:

$$Cont(K) = 1 - \frac{2}{NK(2N - 3K - 1)} \sum_{i=1}^N \sum_{x_j \in V_K(x_i)} (\hat{r}(x_i, x_j) - K), \tag{8}$$

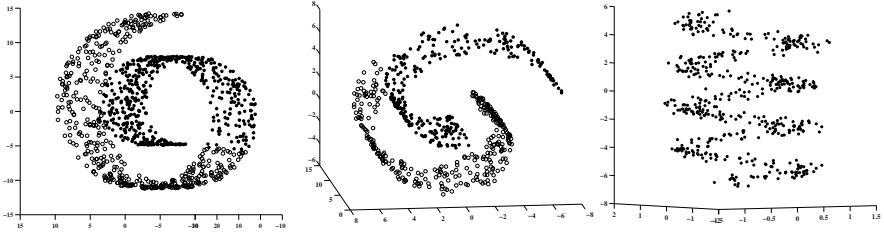


Fig. 1. The three datasets used in the experiments. (left): *Swiss-Roll*, where two contiguous fragments are identified with different symbols in order to check manifold contiguity preservation in Fig. 2 (middle): *Two-Spirals*, again with different symbols for each of the spiral fragments. (right): *Helix*.

where $V_K(x_i)$ is the set of data points x_j for which $x_j \notin \hat{C}_K(x_i) \wedge x_j \in C_K(x_i)$. The terms $r(x_i, x_j)$ and $\hat{r}(x_i, x_j)$ are the ranks of x_j when data points are ordered according to their distance from the data vector x_i in the observed data space and in the low-dimensional representation space, respectively, for $i \neq j$.

4.2 Datasets

Three artificial 3-dimensional datasets, represented in Fig. 1, were used in the experiments that follow. The first one is the *Swiss-Roll* dataset, consisting on 1000 randomly sampled data points generated by the function: $(x_1, x_2) = (t \cos(t), t \sin(t))$, where t follows a uniform distribution $\mathcal{U}(3\pi/2, 9\pi/2)$ and the third dimension follows a uniform distribution $\mathcal{U}(0, 21)$. The second dataset, herein called *Two-Spirals*, consists of two groups of 300 data points each that are similar to *Swiss-Roll* although, this time, the first group follows the uniform distribution $\mathcal{U}(3\pi/4, 9\pi/4)$, while the second group was obtained by rotating the first one by 180 degrees in the plane defined by the first two axes and translating it by 2 units along the third axis. The third dataset, herein called *Helix*, consists of 500 data points that are images of the function $\mathbf{x} = (\sin(4\pi t), \cos(4\pi t), 6t - 0.5)$, where t follows $\mathcal{U}(-2, 2)$. Gaussian noise of zero mean and standard deviation (σ) of 0.05 was added to *Swiss-Roll* and *Two-Spirals*. A higher level of Gaussian noise ($\sigma = 0.20$) was added to *Helix*.

4.3 Results and Discussion

The goal of the experiments is twofold. Firstly, we aim to assess whether the proposed Geo-GTM model could capture and visually represent the underlying structure of datasets of smooth but convoluted geometry better than the standard GTM. Secondly, we aim to evaluate and compare the trustworthiness and the continuity of the mappings generated by GTM and Geo-GTM.

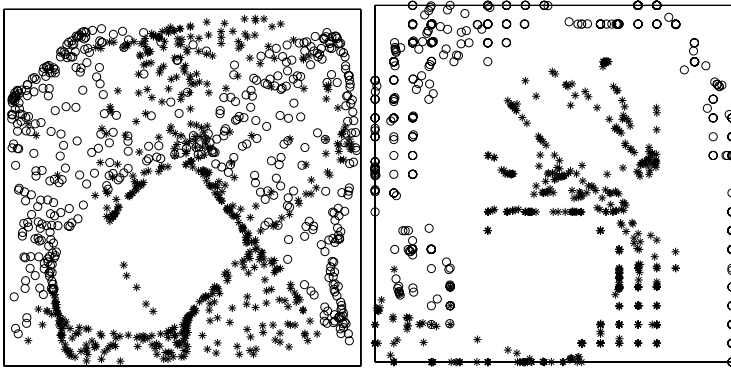


Fig. 2. Data visualization maps for the *Swiss-Roll* set. (Left): standard GTM; (right): Geo-GTM.

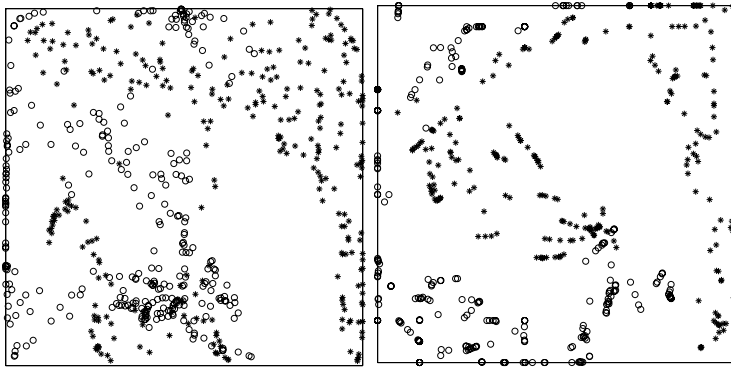


Fig. 3. Visualization maps for the *Two-Spirals* set. (Left): standard GTM; (right): Geo-GTM.

The posterior mean distribution visualization maps for all datasets are displayed in Figs. 2 to 4. Geo-GTM, in Fig. 2, is shown to capture the spiral structure of *Swiss-Roll* far better than standard GTM, which misses it at large and generates a poor data visualization with large overlapping between non-contiguous areas of the data. A similar situation is reflected in Fig. 3. The two segments of *Two-Spirals* are neatly separated by Geo-GTM, whereas the standard GTM suffers a lack of contiguity of the segment represented by circles as well as overlapping of part of the data of both segments. The results are even more striking for *Helix*, as shown in Fig. 4: the helicoidal structure is neatly revealed by Geo-GTM, whereas it is mostly missed by the standard GTM. The former also faithfully preserves data continuity, in comparison to the breaches of continuity that hinder the visualization generated by the latter. Remarkably, Geo-GTM is much less affected by noise than the standard GTM, as it recovers,

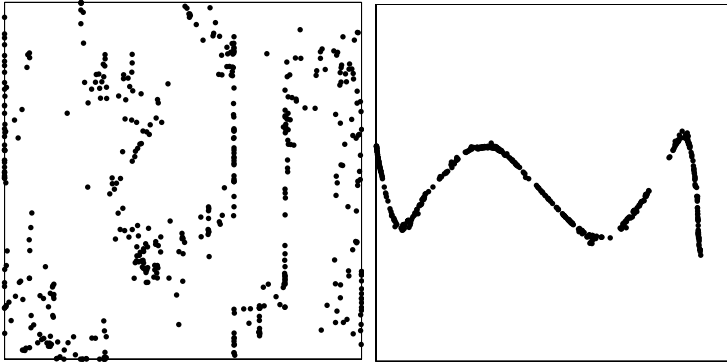


Fig. 4. Data visualization maps for the *Helix* set. (Left): standard GTM; (right): Geo-GTM.

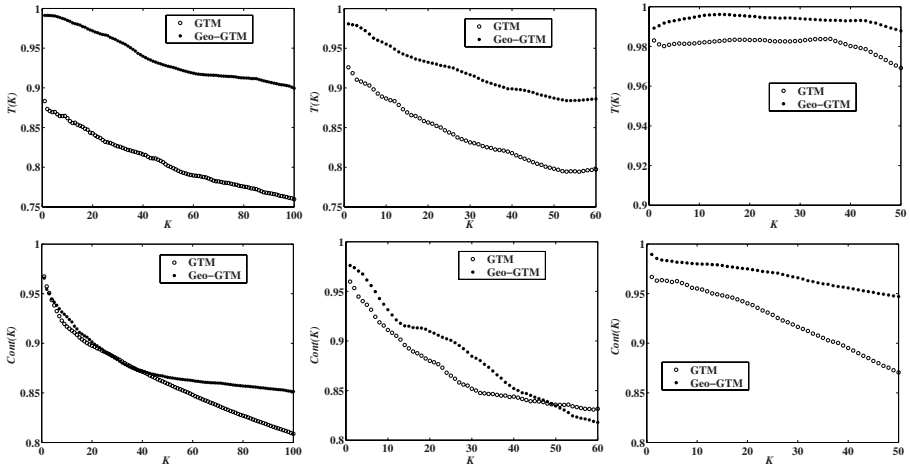


Fig. 5. Trustworthiness (top row) and continuity (bottom row) for (left column): *Swiss-Roll*, (middle column): *Two-Spirals*, and (right column): *Helix*, as a function of the neighbourhood size K

in an almost perfect manner, the underlying noise-free helix. This is probably due to the fact that this model favours directions along the manifold, minimizing the impact of off-manifold noise.

The trustworthiness and continuity for all datasets are shown in Fig. 5. As expected from the visualization maps in Figs. 2-4, the Geo-GTM mappings are far more trustworthy than those generated by GTM for neighbourhoods of any size across the analyzed range. The differences in continuity preservation are smaller although, overall, Geo-GTM performs better than GTM model, specially with the noisier *Helix* dataset.

5 Conclusion

In this brief paper, we have introduced a variation of the manifold learning GTM model, called Geo-GTM, and shown that it is able to faithfully recover and visually represent the underlying structure of datasets of smooth but convolute geometries. It does so by limiting the effect of manifold folding through the penalization of the discrepancies between inter-point Euclidean distances and the approximation of geodesic distances along the model manifold.

The reported experiments also show that the mappings generated by Geo-GTM are more trustworthy than those generated by the standard GTM, while preserving continuity better. Moreover, Geo-GTM has been shown to recover the true underlying data structure even in the presence of noise. This capability of the model should be investigated in detail in future research, using both synthetic and real data sets.

Acknowledgements. Alfredo Vellido is a researcher within the Ramón y Cajal program of the Spanish MICINN and acknowledges funding from the CICYT research project TIN2006-08114. Raúl Cruz-Barbosa acknowledges SEP-SESIC (PROMEP program) of México for his PhD grant.

References

1. Figueiredo, M.A.T., Jain, A.K.: Unsupervised learning of finite mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24(3), 381–396 (2002)
2. Bishop, C.M., Svensén, M., Williams, C.K.I.: The Generative Topographic Mapping. *Neural Computation* 10(1), 215–234 (1998)
3. Vellido, A.: Missing data imputation through GTM as a mixture of t-distributions. *Neural Networks* 19(10), 1624–1635 (2006)
4. Vellido, A., Lisboa, P.J.G., Vicente, D.: Robust analysis of MRS brain tumour data using t-GTM. *Neurocomputing* 69(7-9), 754–768 (2006)
5. Archambeau, C., Verleysen, M.: Manifold constrained finite Gaussian mixtures. In: Cabestany, J., Gonzalez Prieto, A., Sandoval, F. (eds.) *IWANN 2005*. LNCS, vol. 3512, pp. 820–828. Springer, Heidelberg (2005)
6. Tenenbaum, J.B., de Silva, V., Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction. *Science* 290, 2319–2323 (2000)
7. Lee, J.A., Lendasse, A., Verleysen, M.: Curvilinear Distance Analysis *versus* Isomap. In: *Proceedings of European Symposium on Artificial Neural Networks (ESANN)*, pp. 185–192 (2002)
8. Bernstein, M., de Silva, V., Langford, J., Tenenbaum, J.: Graph approximations to geodesics on embedded manifolds. Technical report, Stanford University, CA (2000)
9. Dijkstra, E.W.: A note on two problems in connection with graphs. *Numerische Mathematik* 1, 269–271 (1959)
10. Venna, J., Kaski, S.: Neighborhood preservation in nonlinear projection methods: An experimental study. In: Dorffner, G., Bischof, H., Hornik, K. (eds.) *ICANN 2001*. LNCS, vol. 2130, pp. 485–491. Springer, Heidelberg (2001)

Guaranteed Network Traffic Demand Prediction Using FARIMA Models

Mikhail Dashevskiy and Zhiyuan Luo

Computer Learning Research Centre,
Royal Holloway, University of London
Egham, Surrey TW20 0EX, UK
mikhail@cs.rhul.ac.uk

Abstract. The Fractional Auto-Regressive Integrated Moving Average (FARIMA) model is often used to model and predict network traffic demand which exhibits both long-range and short-range dependence. However, finding the best model to fit a given set of observations and achieving good performance is still an open problem. We present a strategy, namely Aggregating Algorithm, which uses several FARIMA models and then aggregates their outputs to achieve a guaranteed (in a sense) performance. Our feasibility study experiments on the public datasets demonstrate that using the Aggregating Algorithm with FARIMA models is a useful tool in predicting network traffic demand.

1 Introduction

Prediction using time series models has many important applications. One of these applications is network traffic demand prediction where having observed network traffic demand in the past (for example, each hour for a finite period of time), we want to predict the future demand. Successful network traffic demand prediction can be used to improve the Quality of Service, to route packages in a more efficient way or to solve some other problems in a network. It has been shown that network traffic demand typically has both long-range and short-range dependence ([13]). Fractional Auto-Regressive Integrated Moving Average (FARIMA) model has been proposed for modeling and prediction of such network traffic demand ([3],[4],[9]) and fairly good results can be achieved. However, it is still an open problem on how to find the best possible model for a given set of observations.

In this paper, we suggest a different approach by considering the problem of prediction with expert advice. In this setting we have a set of models (i. e. experts) and instead of finding the best possible model for a dataset and predicting absolutely well, we aim to perform, on average, almost as well as the overall best expert. In particular, we describe one optimal strategy, namely Aggregating Algorithm ([11]) to mix predictions of the experts (models' outputs) so that there is a guaranteed upper bound on the regret term (the difference between the loss of the algorithm and the loss of the best expert). Therefore, the proposed approach is a reliable method in minimizing risks of having bad performances of a prediction system.

The rest of the paper is organized as follows. Section 2 briefly describes the FARIMA model and its parameter estimation. Section 3 describes Aggregating Algorithm and Section 4 discusses the experimental setup and experimental results. Finally Section 5 presents a conclusion and possible directions for further research.

2 Fractional Auto-Regressive Integrated Moving Average (FARIMA)

For the network traffic demand prediction problem, we have observations $x_1, x_2, \dots; x_i \in \mathbb{R} \forall i$ and try to find a sequence of functions: $f_n : x_1, \dots, x_n \rightarrow x_{n+1}$, such that its predictions are close to real observed values (in a sense). This problem is naturally related to the discrete time series analysis.

In this section we briefly describe FARIMA, which was introduced in [4,5] and has been used to model and predict processes with long-range dependence, such as network traffic demand, electricity demand and financial data (when we can assume that the variance is constant). The FARIMA model is a generalization of the Auto-Regressive Moving Average (ARMA) model which works well with processes having short-range dependence and linear dependence between the observations. We assume that FARIMA model can model network traffic demand well (with appropriately chosen parameters) as there exist a number of research papers ([13,14]) supporting this statement.

Consider a time series process $\{X_t, t \in \mathbb{Z}\}$ which is weakly stationary (has a finite mean and the covariance depends only on the difference between the parameters).

Definition 1. *The autocovariance function of process $\{X_t\}$ is $\gamma(s, t) = E[(X_s - \mu)(X_t - \mu)]$, where $\mu = E X_t$.*

As for weakly stationary processes the covariance depends only on the difference between two parameters we can write $\gamma(s, t) = E[(X_s - \mu)(X_t - \mu)] = E[(X_0 - \mu)(X_{t-s} - \mu)] = \gamma(t-s)$. We consider long-range dependent time series processes:

Definition 2. *The time series process $\{X_t, t \in \mathbb{Z}\}$ is called long-range dependent if $\sum_{h=-\infty}^{\infty} |\gamma(h)| = \infty$.*

It is often assumed that the autocovariance function of a long-range dependent process can be written in the form $\gamma(h) \sim h^{2d-1}l_1(h)$, where $l_1(h)$ is a slowly varying function and d reflects the order of long-range dependency.

Formally a FARIMA process $\{X_t\}$ can be defined by

$$\phi(B)X_t = \theta(B)(1 - B)^{-d}\epsilon_t,$$

where B is the backshift operator defined as $B^k X_t = X_{t-k}$, $\phi(B) = 1 + \phi_1 B + \dots + \phi_p B^p$ and $\theta(B) = 1 + \theta_1 B + \dots + \theta_q B^q$ are the autoregressive and moving

average operators respectively; $\phi(B)$ and $\theta(B)$ have no common roots, $(1 - B)^{-d}$ is a fractional differentiating operator defined by binomial expansion

$$(1 - B)^{-d} = \sum_{j=0}^{\infty} \eta_j B^j = \eta(B), \text{ where } \eta_j = \frac{\Gamma(j + d)}{\Gamma(j + 1)\Gamma(d)}$$

for $d < \frac{1}{2}, d \neq 0, -1, -2, \dots$, and where $\Gamma(z) = \int_0^{\infty} t^{z-1} e^{-t} dt$ is the Gamma function (see [7,2]) and ϵ is a white noise sequence with finite variance. We call such process $\{X_t\}$ a FARIMA(p, d, q) process.

Parameter estimation is the first step in fitting a FARIMA model (to some data), which can then be used to predict next values of the time series. The parameter estimation problem for FARIMA involves estimation of d which describes the long-range dependence and estimation of $\phi(B)$ and $\theta(B)$ which describe the short-range dependence. There exist different techniques to estimate the parameters associated with a FARIMA process. In this paper, we assume that the knowledge of the model orders p and q is available. Parameter d is related to the Hurst parameter H via the equation $H = d + \frac{1}{2}$. The following procedure is used to estimate parameter d and the polynomial coefficients of $\phi(B)$ and $\theta(B)$ with fixed parameters p and q :

1. Estimate the parameter $d = H - 0.5$, where H is Hurst parameter and is estimated using Dispersional analysis (also known as the Aggregated Variance method), see [10,1] for more details.
2. Differentiate the data with the estimated parameter d .
3. Use the Innovations Algorithm ([2]) to fit a Moving Average model into the data.
4. Estimate the coefficients of $\phi(B)$ and $\theta(B)$ ([2]).

3 Aggregating Algorithm

FARIMA models can efficiently represent time series data which exhibit both long-range and short-range dependence. However, it is still an open question on how to fit the best model given the data. To overcome the difficulty of choosing the best parameters of a model, we propose a new approach and describe a strategy to mix outcomes of several models according to their individual losses. Our aim is not to find the best model predicting network traffic demand, but to find an efficient way to use good models so that we can have a guaranteed performance.

This problem has been studied extensively in the area of prediction with expert advice and can be described in the form of a game. In the game, there are experts who can give advice on predictions of the possible outcome and we do not know which expert can perform well in advance. At each round of the game, experts give their predictions based on the past observations and we can study the experts' predictions in order to make our prediction. Once our prediction is made, the outcome of event is known. As a result, the losses suffered by each

expert and our algorithm are calculated. Our aim is to find a strategy which can guarantee the total loss (i. e. the cumulative loss on all rounds of the game) is not much worse than the total loss of the best expert during the game.

Now we will give a more formal definition of the prediction with expert advice. Considering a game between three players, Nature, Experts, and Learner, let Σ be a set called *signal space*, Ω be a set called *sample space*, Γ be a set called the *decision space*, and Θ be a measurable space called the *parameter space*. A *signal* is an element of the signal space, an *outcome* is an element of the sample space and a *decision* is an element of the decision space. The *loss function* $\lambda : \Omega \times \Gamma \rightarrow [0, \infty]$ which measures the performance of Experts and Learner is also part of the game. At each round $T = 1, 2, \dots$, the perfect-information game is described below.

1. Nature outputs a signal $x_T \in \Sigma$.
2. Experts make measurable predictions $\xi_T : \Theta \rightarrow \Gamma$; where $\xi_T(\theta)$ is the prediction corresponding to the parameter $\theta \in \Theta$.
3. Learner makes his own prediction $\gamma_T \in \Gamma$.
4. Nature chooses an outcome $\omega_T \in \Omega$.
5. Experts' loss calculation $L_T(\theta) = \sum_{t=1}^T \lambda(\omega_t, \xi_t(\theta))$.
6. Learner's loss calculation $L_T = \sum_{t=1}^T \lambda(\omega_t, \gamma_t)$.

Learner wants to minimize the regret term, i. e. the difference between the total loss of the algorithm and the total loss of the best expert. Many strategies have been developed for Learner to make his prediction γ_T such as following the perturbed leader and gradient descent. In this paper we consider the case $\Omega \subset \mathbb{R}, \Gamma \subset \mathbb{R}$ and present one of the methods solving the problem, namely Aggregating Algorithm (AA). This algorithm has many theoretical advantages ([11]), one of which is a guaranteed optimal performance (which cannot be improved).

Aggregating Algorithm works as follows: at each step t it re-computes weights of the experts (represented by their probability distribution) and mixes all experts' predictions according to this distribution. Thus the AA gets a mixture of the experts' predictions (a *generalized prediction*), then it finds the best (in some sense) prediction from Θ . There are two parameters in the AA: learning rate is $\eta > 0$ (β is defined to be $e^{-\eta}$) and P_0 is a probability distribution on set Θ of experts. Intuitively P_0 is the prior distribution, which specifies the initial weights assigned to the experts. In this way, at each step t of the algorithm the AA performs the following actions to make the prediction at the step 3 in the above game:

3.1 Updating experts' weights according to the previous losses:

$$P_{t-1}(d\theta) = \beta^{\lambda(\omega_{t-1}, \xi_{t-1}(\theta))} P_{t-2}(d\theta), \quad \theta \in \Theta.$$

3.2 Predictions' mixing according to their weights:

$$g_t(\omega) = \log_{\beta} \int_{\Theta} \beta^{\lambda(\omega, \xi_t(\theta))} P_{t-1}^*(d\theta), \tag{1}$$

where $P_{t-1}^*(d\theta)$ is a normalised probability distribution, i. e. $P_{t-1}^*(d\theta) = \frac{P_{t-1}(d\theta)}{P_{t-1}(\Theta)}$ and if $P_{t-1}(\Theta) = 0$, AA is allowed to choose any prediction. On this step we get $g_t(\omega)$ which is a function $\Omega \rightarrow \mathbb{R}$.

3.3 Looking for an appropriate prediction from $\Gamma : \gamma_t(g_t)$.

In [11] (Lemma 2) it is proved that in the case of the *square-loss function*, i. e. $\lambda(\omega, \gamma) = (\omega - \gamma)^2$, $\eta \leq \frac{1}{2Y}$ (where Y is a bound for $|y_t|$)

$$\gamma_t = \frac{g_t(-Y) - g_t(Y)}{4Y} \quad (2)$$

is a prediction from Θ .

The formula (2) gives us the final prediction at step t . In our experimental setting experts correspond to FARIMA models and $\xi_t(\theta)$ corresponds to the prediction at step t given by a specific FARIMA model denoted by θ . As we use only a finite number of experts the integration in the formula (1) is now a sum. Preprocessing of the datasets can ensure $|Y| = 1$.

In this paper we consider square loss. Assuming that the number of experts is finite, it has been proved ([12]) that the accumulative loss of the AA $L_T(A)$ is bounded by for all round T of the game and each expert θ as $L_T(AA) \leq L_T(\theta) + \ln(n)$ where n is the total number of experts.

4 Experiments and Results

In this section we describe the experiments carried out to evaluate the performance of the AA. Two publicly available datasets are used ([8,6]). For illustration purpose, the measurement time interval is set to 1 second on these two datasets and we use only relatively small extracts of these datasets: from each of them two extracts of 360 entries are obtained and there are now four sets for our experiments, namely A, B, C and D. Each of these four sets is divided into two parts of 180 observations. The first part (training set) is used for fitting a FARIMA model and estimating the corresponding parameters. The second part (test set) is used to run experiments in the online mode, i. e. after making a prediction, we receive the true value of network traffic demand. In our experiments the datasets are preprocessed by the following operations: subtraction of the mean value and then division by the absolutely maximum entry.

In the experiments, the FARIMA models with different sets of parameters p and q (since the parameter d is estimated with a method which we consider reliable) are experts and a finite number of FARIMA models are considered in the AA. It has been suggested that the orders p and q of a FARIMA model are relatively small. Therefore, we use the models with parameters $0 \leq p, q \leq 3, p + q \neq 0$ (altogether we used 15 experts — FARIMA models). The prior distribution on the all 15 experts used in the AA is chosen to be uniform. The learning rate η of the AA is set to be $\frac{1}{2}$. During the experiments, the window size is not fixed so that at each step of the algorithm we use all the previous observations.

Table 1. Experimental results (accumulated square loss)

Algorithm	Dataset A	Dataset B	Dataset C	Dataset D
$FARIMA_{best}$ parameter (p,q)	7.40 (3, 0)	5.79 (0, 3)	3.23 (1, 1)	15.85 (1, 0)
$FARIMA_{worst}$ parameter (p,q)	125.38 (1, 2)	28.92 (1, 2)	158.35 (3, 1)	102.65 (2, 2)
AA – FARIMA	7.83	5.90	3.39	15.92
Number of experts better than AA	2	8	9	1
Number of experts worse than AA	13	7	6	14
Avr. diff. between good experts and AA	0.39	0.05	0.07	0.08
Avr. diff. between of bad experts and AA	14.45	11.86	42.82	14.73
$Mean_{best}$ parameter	12.25 1	9.62 15	5.22 5	18.41 1
$Mean_{worst}$ parameter	20.17 56	12.54 1	6.27 41	25.15 49
Constant	18.68	10.63	5.67	23.62

To compare the performance of the FARIMA model with other methods we consider the naive algorithm *Mean* which gives the mean value of several previous observations (we also tried the *Median* algorithm which performed slightly worse on all datasets) and *Constant* which gives the mean value of the whole dataset. The square-loss function is used as the metric to compare their performance.

The summary of the experimental results is shown in Table 1. It can be seen that AA-FARIMA performs not much worse than the best model (expert) and the differences between the total losses of the best experts and the worst experts are large. The experimental results show that the FARIMA model outperformed simple methods Mean and Constant. For each dataset, all the experts can be divided into two categories: good experts (the experts performing better than the AA) and bad experts (the rest of the experts). We can see that for two out of four datasets (i. e. datasets A and D) there are only 1 or 2 good experts and for the rest of the datasets the experts are divided almost equally between the two groups. In all cases, however, the total loss of the AA is very close to the average total loss of the good experts, but are significantly smaller than the average total loss of the bad experts. It means that if we are unlucky to use one of the bad experts as our prediction model then the AA outperforms it significantly. On the other hand, if we choose a good expert as the prediction model, then the AA is not much worse than this model.

Figure 1 shows the performance of the two algorithms, namely the best FARIMA model and AA-FARIMA on dataset D. The best FARIMA model predicts the peaks but gives absolutely smaller values than the real demand as the algorithm is not confident in its predictions. The most interesting trend which can be seen in this figure is that AA-FARIMA at the beginning differs slightly from the best FARIMA model, but towards the end of the experiment it gives practically the same predictions as the best FARIMA model, which is the best

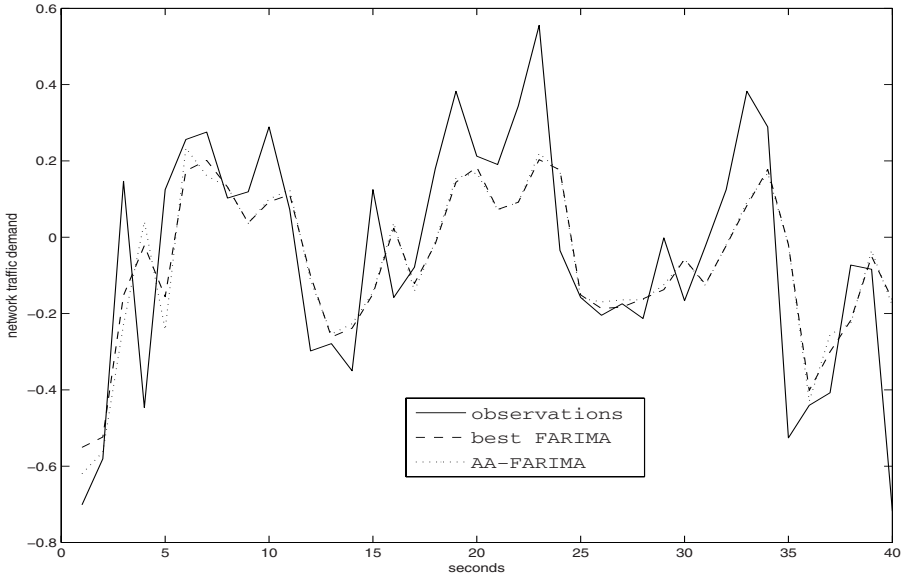


Fig. 1. Observed and predicted network traffic demand for dataset D (showing first 40 observations)

expert in this example. It means that the AA is able to recognize the best expert and follow its predictions effectively.

5 Conclusions

FARIMA processes have been used with some success for modeling and prediction of the network traffic with both short and long range dependence. However, it is still an open problem on how to find the best possible model for a set of observations and achieve good performance. The experimental results showed that choosing wrong parameters of FARIMA models can lead to worse prediction performance. To address the issue of guaranteed performance of using the FARIMA model, we considered the framework of prediction with expert advice and presented a strategy, namely Aggregating Algorithm which can minimize the risk of choosing wrong parameters of the model at the cost of a slightly higher total loss comparing to that of the best expert. We believe that it is worth tolerating some additional loss in order to be more confident in the performance of the algorithm. In other words, its performance will not differ too much from that of the best model. The experiments demonstrated the usefulness of the AA in predicting network traffic demand with a performance guarantee.

One of other advantages of the AA is that it can deal with the cases where the best model can be different at different times. It means that if the parameters of the process change over time and the model which was the best at the beginning

does not perform well any more, then its weight in the AA will be decreased over the time and eventually its prediction will practically not be taken into account in prediction. We are planning to carry out experiments to explore this feature.

Acknowledgements

This work has been supported by the EPSRC through grant EP/E000053/1 “Machine Learning for Resource Management in Next-Generation Optical Networks”.

References

1. Blok, H.J.: On The Nature Of The Stock Market: Simulations And Experiments. PhD Thesis, University of British Columbia, Canada (2000)
2. Brockwell, P.J., Davis, R.A.: Time Series: Theory and Methods. Springer, Heidelberg (1991)
3. Dethe, C.G., Wakde, D.G.: On the prediction of packet process in network traffic using FARIMA time-series model. *J. of Indian Inst. of Science* 84, 31–39 (2004)
4. Granger, C.W.J., Joyeux, R.: An introduction to long-memory time series models and fractional differencing. *J. of Time Series Analysis* 1, 15–29 (1980)
5. Hosking, J.R.M.: Fractional differencing. *Biometrika* 68, 165–176 (1981)
6. Leland, W.E., Taqqu, M.S., Willinger, W., Wilson, D.V.: On the self-similar nature of Ethernet traffic. *IEEE/ACM Trans. on Networking*. 2, 1–15 (1994)
7. Palmo, W.: Long-Memory Time Series. Theory and Methods. Wiley Series in Probability and Statistics (2007)
8. Paxson, V.: Fast Approximation of Self-Similar Network Traffic. Technical report LBL-36750/UC-405 (1995)
9. Shu, Y., Jin, Z., Zhang, L., Wang, L., Yang, O.W.W.: Traffic prediction using FARIMA models. In: *IEEE International Conf. on Communication*, vol. 2, pp. 891–895 (1999)
10. Taqqu, M.S., Teverovsky, V., Willinger, W.: Estimators for long-range dependence: An empirical study. *Fractals* 3, 785–788 (1995)
11. Vovk, V.: Competitive On-line Statistics. *Int. Stat. Review* 69, 213–248 (2001)
12. Vovk, V.: Prediction with expert advice for the Brier game (2008), <http://arxiv.org/abs/0710.0485>
13. Willinger, W., Paxson, V., Riedi, R.H., Taqqu, M.S.: Long-range dependence and data network traffic. *Theory And Applications Of Long-Range Dependence*, 373–407 (2003)
14. Xue, F., Liu, J., Zhang, L., Yang, O.W.W.: Traffic Modelling Based on FARIMA Models. In: *Proc. IEEE Canadian Conference on Electrical and Computer Eng.* (1999)

A New Incremental Algorithm for Induction of Multivariate Decision Trees for Large Datasets

Anilu Franco-Arcega¹, J. Ariel Carrasco-Ochoa¹, Guillermo Sánchez-Díaz²,
and J. Fco Martínez-Trinidad¹

¹Computer Science Department

National Institute of Astrophysics, Optics and Electronics

Luis Enrique Erro # 1, Santa Maria Tonantzintla, Puebla, Mexico, C.P.72840

{anifranco6, ariel, fmartine}@inaoep.mx

²Centro Universitario de los Valles

Universidad de Guadalajara

Carretera Guadalajara - Ameca Km. 45.5, Ameca, Jalisco, Mexico, C.P. 46600

guillermo.sanchez@profesores.valles.udg.mx

Abstract. Several algorithms for induction of decision trees have been developed to solve problems with large datasets, however some of them have spatial and/or runtime problems using the whole training sample for building the tree and others do not take into account the whole training set. In this paper, we introduce a new algorithm for inducing decision trees for large numerical datasets, called IIMDT, which builds the tree in an incremental way and therefore it is not necessary to keep in main memory the whole training set. A comparison between IIMDT and ICE, an algorithm for inducing decision trees for large datasets, is shown.

Keywords: Decision trees, supervised classification, large datasets.

1 Introduction

The main objective of a supervised classification algorithm is to determine the class of a new object (described by an attribute set) based on the information contained in a set of previously classified objects (training set — TS). There are different algorithms for solving the supervised classification problem [1] like: statistical algorithms, distance-based algorithms, neural networks, decision trees, etc. Decision trees are very popular in Data Mining [2], because of their simple construction.

A decision tree (DT) is a tree structure formed by nodes (internal nodes and leaves) and edges. Internal nodes are characterized by one or more test attributes and each node has one or more children. Each one of the edges has a value for the test attribute, this value determines the path to be followed in the tree. On the other hand, leaves contain information that allows to determine the class of an object. To classify a new object, it traverses the tree starting from the root until a leaf is reached, when the object arrives at a leaf it is classified according to the information stored in that leaf.

There are several algorithms for inducing a DT [3,4,5,6,7]. However, not all of them can handle large datasets, because they need to keep in main memory the whole training set for building the DT [8]. Some algorithms for induction of DT have been developed to work with large datasets [9,10,11,12,13,14], however the required space to store the data in some of them is at least double the space required for the whole training set, and in some other algorithms the DT is built based only on a small subset of the training objects, but this subset could be not representative of the whole training set.

In this paper we introduce a new Incremental Induction of Multivariate DT algorithm for large datasets (IIMDT), which consider the whole training set for building the tree but it is not stored in main memory. IIMDT processes the training objects one by one, in this way only the processed object must be kept in main memory at each step.

The rest of this work is organized as follows: Section 2 presents some works related to DT induction algorithms for large datasets, in Section 3 the proposed algorithm is introduced, Section 4 shows the experimental results and a comparison against other DT generation algorithms. Finally in Section 5 conclusions and future work are presented.

2 Related Work

Currently, there are several algorithms for inducing DT for large datasets. SLIQ [9] solves the problem of storing in main memory the whole training set representing the attributes by lists that store the values of the attributes, for each object, these lists can be stored in disk. Additionally, SLIQ uses a list that contains the class of each object and the number of the node where this object is stored in the tree. However, this last list must be stored in main memory, which could be a problem for large datasets, because the size of the list depends on the number of objects in the training set. SLIQ sorts the numerical attributes before building the DT, therefore it is not necessary to carry out this procedure when each node is expanded.

SPRINT [10] is an improvement of SLIQ, the difference lies in how SPRINT represents the lists for each attribute. This algorithm adds a column to each list for storing the class of each object, hence SPRINT does not need to store in main memory any whole list, however, since it has to read all the lists for expanding each node, the processing time could be too large if the training set has a lot of objects.

CLOUDS [11] simplifies SPRINT, representing the numerical attributes by intervals, in this way CLOUDS reduces substantially the required time for choosing the attributes that will represent the internal nodes of the DT. A drawback of SPRINT and CLOUDS is that they require at least double the required space for storing the training set.

RainForest [12] follows the idea of representing the attributes by lists, nevertheless it tries to reduce these lists storing only all the different values for each attribute, thus the list size will not be of the number of training objects but

equal to the number of different values. These lists are stored in main memory, therefore if the attributes had a lot of different values in the training set, the available space could not be enough.

BOAT [13] is an incremental algorithm that avoids to store the whole training set in main memory using only an object subset for building the DT. Starting from this subset BOAT applies the bootstrapping technique for generating multiple DT and combining them. Afterwards BOAT verifies the whole training set to refine the constructed DT.

ICE [14] is another incremental algorithm that also is based on a subset for building the DT. This algorithm divides the training set in subsets called epochs, for each epoch ICE builds a DT, which is used to obtain a subset of objects. These subsets, obtained from each epoch, are joined for building the final DT. ICE does not guarantee that the obtained subset is representative of the whole training set.

3 IIMDT Algorithm

In this section we introduce IIMDT, an incremental algorithm for building a DT for large datasets. Following the idea proposed in [7], IIMDT builds a multivariate DT considering all the attributes for the internal nodes, therefore our algorithm does not access the training set for choosing test attributes in each node expansion. Our algorithm processes the objects of the training set one by one, in this way it does not need to keep in memory the whole training set to generate the DT, but only the object that is being processed.

To build the DT, the proposed algorithm begins with a root node initially empty and without descendants. Each object of the training set will traverse the tree until a leaf is reached, in which the object will be stored. When a leaf has s stored objects there are two cases:

1. If the leaf contains objects from more than one class, it will be expanded generating one edge for each class of objects in the node. Each generated edge will lead to an empty leaf and the attributes values associated to each edge will be taken from a representative object obtained from the objects in the node that belong to the class of the edge. Once the attributes values are obtained, the s stored objects are deleted from the node. In order to allow the nodes to be expanded having objects from different classes, IIMDT reorganizes the training set alternating objects from each class. This reorganization process is very fast; in the experiments the time required for this reorganization will be included as part of the IIMDT runtime.
2. If the leaf has objects belonging to the same class, it will not be expanded, but the attributes values associated to the representative object obtained from the stored objects are combined with attributes values associated to the input edge of the leaf. The s stored objects in the leaf are deleted and the counter of objects in the leaf is restarted to 0.

To traverse the tree an object O starts at the root. When O arrives to an internal node, O follows the path of the edge that best matches its attribute

values. When all the objects had been processed for building the tree, IIMDT assigns to each leaf the majority class of objects stored in that leaf. The IIMDT algorithm is as follows:

Input: TS (training set), s
Output: DT (decision tree)
Step 1: Reorganize $TS(TS)$
Step 2: $ROOT \leftarrow \text{CreateNode}()$
Step 3: For each $O_i \in TS$, do
 Update $DT(O_i, ROOT)$
Step 4: AssignClassToLeaves()

Update $DT(O_i, ROOT)$ is as follows:

UpdateDT ($O_i, NODE$)
 If $NODE.numObj < s$, then
 AddObjNode($NODE, O_i$)
 $NODE.numObj = NODE.numObj + 1$
 If $NODE.numObj = s$, then
 ExpandNode($NODE$)
 $NODE.numObj = NODE.numObj + 1$
 Else /* $NODE.numObj > s$ */
 For each edge $R_j \in NODE$, do
 $sim_i[j] = \text{Similarity}(O_i, NODE.OR_j)$
 $Edge = \max_i(sim_i[i])$
 UpdateDT ($O_i, NODE.Edge$)

Similarity($O_i, NODE.OR_j$) computes the similarity between the object O_i and the representative object of the edge j ; *ExpandNode*($NODE$) is as follows:

ExpandNode ($NODE$)
 If $NODE.classes > 1$
 For each class $C_i \in NODE$, do
 $R_i \leftarrow \text{CreateEdge}()$
 $OR_i = \text{ObtainRepObj}(C_i, NODE)$
 For each edge $R_i \in NODE$, do
 $leaf_i \leftarrow \text{CreateNode}()$
 Delete($NODE.Obj$)
 Else
 $OR = \text{ObtainRepObj}(C, NODE)$
 Combine($OR, NODE.OR$)
 Delete($NODE.Obj$)
 $NODE.numObj = 0$

Once the DT have been constructed, it can be used to classify unseen objects traversing the tree until a leaf is reached and assigning the class associated to that leaf.

4 Experimental Results

To show the performance of IIMDT, experiments with four datasets, described in Table 1, were done. All our experiments were performed on a Pentium 4 at 3.06 GHz, with 2 GB of RAM running Kubuntu 7.10. In all the experiments, the mean was used to obtain the representative objects and for combining attribute values the average between them was used. The value of s was determined experimentally testing over the datasets of Table 1, different values for s from 50 to

Table 1. Description of the dataset used in the experiments

Dataset	# Classes	# Objects	# Attributes
Letter [15]	26	20,000	16
Poker [15]	10	1,000,000	10
SpecObj* [16]	6	884,054	5
GalStar* [16]	2	4,000,000	30

Table 2. Results obtained for Letter

Algorithm	Time (sec)	Accuracy rate	Size
IIMDT	2.97	87.6	522.2
ICE	5.42	75.4	946.6
C4.5	6.08	87.4	2163.4

900 with increments of 50 were tested; $s = 100$ yielded the best results, therefore this value was used for the experiments.

In the experiments we evaluated the execution time (including the induction time and the classification time, and for IIMDT also including the reorganization time), the accuracy rate over a testing set, and the number of nodes of the generated DT.

For Letter we used five-fold cross-validation over the whole set. For this dataset, IIMDT was compared against C4.5 and ICE, an algorithm for building DT for large datasets (for our experiments we implemented ICE based on [14]). Since Letter is relatively small we could apply the C4.5 algorithm for comparing the accuracy of the algorithms for large datasets. The results for this dataset are shown in Table 2, in this experiment IIMDT is similar to C4.5 and improves ICE in accuracy, but IIMDT builds a tree with less nodes in a shorter time.

Using Poker, SpecObj and GalStar datasets the performance of the DT algorithm IIMDT when the size of the training set increases was evaluated. For these datasets we compared IIMDT against ICE. For Poker and SpecObj we used a test set with 400,000 objects and with the remaining objects of the dataset we created different-size training sets (20,000, 40,000, from 50,000 to 450,000 with increments of 50,000, and for the Poker dataset also a 500,000-object training set).

Fig. 1 shows the results obtained for Poker. As we can see the processing time spent by IIMDT was lower than ICE's for large datasets and both algorithms obtained similar accuracy. Besides, the DT generated by IIMDT has less nodes than the tree generated by ICE. For example, for the 500,000-object training set, IIMDT generated a DT with 7,281 nodes, while the DT generated by ICE had 11,327 nodes.

Fig. 2 shows the results for SpecObj. Based on these results we noticed that IIMDT improves to ICE in accuracy. Although the processing time of ICE is lower than the IIMDT's, we can observe that the processing time of ICE grows faster than the IIMDT's, therefore for a larger dataset, the time of IIMDT will be lower than the time of ICE. About the number of nodes, we noticed that both IIMDT and ICE build similar DT. For example, for the 20,000-object training set the DT generated by IIMDT had 400 nodes and the DT generated by ICE had 379 nodes.

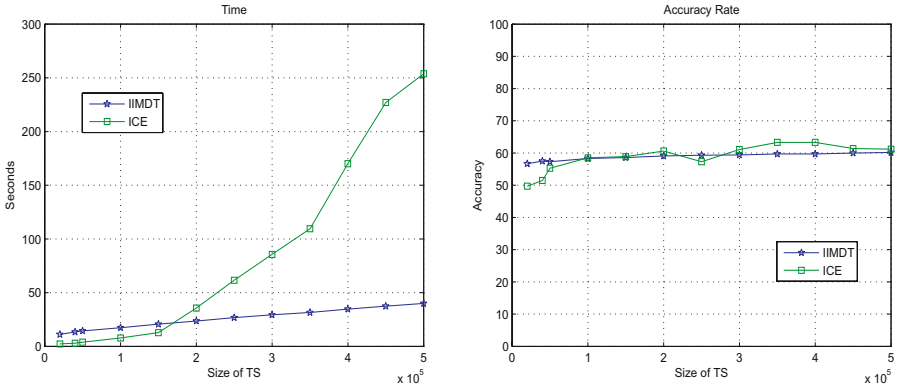


Fig. 1. Processing time and accuracy rate for Poker

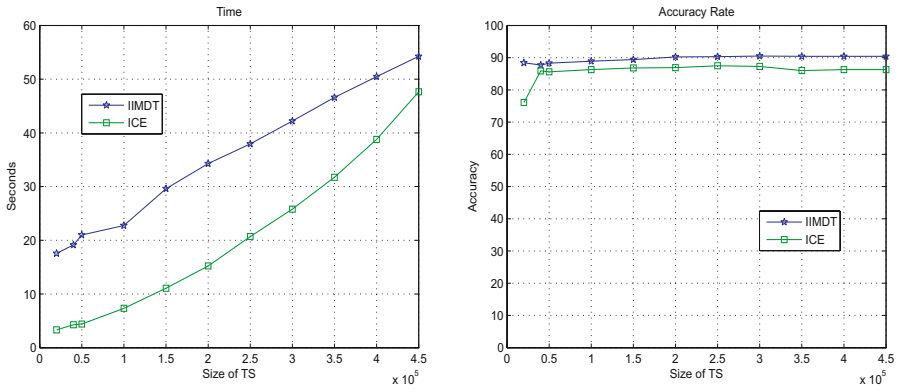


Fig. 2. Processing time and accuracy rate for SpecObj

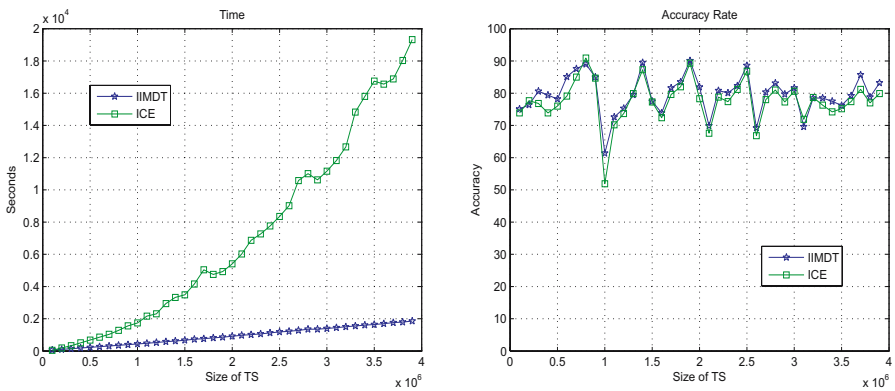


Fig. 3. Processing time and accuracy rate for GalStar

For GalStar we used a test set containing 100,000 objects and with the remaining objects of the dataset we created different-size training sets (from 100,000 to 3,900,000 with increments of 100,000). Fig. 3 shows the results with this dataset. As we can noticed the processing time spent by IIMDT was significantly lower than ICE's and both algorithms obtained similar accuracy. For this dataset, ICE builds trees with less nodes than IIMDT, for example, for the 100,000-training set the generated DT of ICE had 673 nodes while the generated DT of IIMDT had 1393 nodes, however the processing time spent by ICE was higher that IIMDT's.

5 Conclusions and Future Work

Decision trees are a useful tool for solving supervised classification problems, however when we have large datasets the DT construction process could be very expensive. In this paper a new incremental induction of mutivariate DT algorithm called IIMDT was introduced, which builds a DT for large numerical datasets in an incremental way and without storing the whole training set in main memory. The experimental results show that IIMDT builds a DT in a shorter time than ICE, one of the algorithms for inducing decision trees for large datasets, obtaining a similar accuracy rate.

The choice of the parameter s is closely related to the performance of our algorithm, therefore as future work we will study the effect of varying s . Besides we will work on the problem of DT induction for large mixed datasets, because many problems are described by this kind of data.

Acknowledgements. Funding for the SDSS and SDSS-II has been provided by the Alfred P. Sloan Foundation, the Participating Institutions, the National Science Foundation, the U.S. Department of Energy, the National Aeronautics and Space Administration, the Japanese Monbukagakusho, the Max Planck Society, and the Higher Education Funding Council for England. The SDSS Web Site is <http://www.sdss.org/>.

References

1. Dunham, M.: Data Mining, Introductory and Advanced Topics. Prentice Hall, New Jersey (2003)
2. Tan, P., Steinbach, M., Kumar, V.: Introduction to Data Mining. Addison Wesley, Boston (2006)
3. Quinlan, J.R.: C4.5: Programs for Machine Learning. Morgan Kaufmann, San Mateo (1993)
4. Pao, H.-K., Chang, S.-C., Lee, Y.-J.: Model trees for classification of hybrid data types. In: Gallagher, M., Hogan, J.P., Maire, F. (eds.) IDEAL 2005. LNCS, vol. 3578, pp. 32–39. Springer, Heidelberg (2005)
5. Pérez, J., Muguerza, J., Arbelaitz, O., Gurrutxaga, I., Martín, J.: Combining multiple class distribution modified subsamples in a single tree. Pattern Recognition Letters 28(4), 414–422 (2007)

6. Utgoff, P.E.: An improved algorithm for incremental induction of decision trees. In: Proc. 11th International Conference on Machine Learning, pp. 318–325 (1994)
7. Pedrycz, W., Sosnowski: C-fuzzy decision trees. *IEEE Transactions on Systems, Man and Cybernetics - Part C: Applications and reviews* 35(4), 498–511 (2005)
8. Agrawal, R., Imielinski, T., Swami, A.: Database Mining: A Performance Perspective. *IEEE Transactions on Knowledge and Data Engineering* 5(6), 914–925 (1993)
9. Mehta, M., Agrawal, R., Rissanen, J.: SLIQ: A fast scalable classifier for data mining. In: Apers, P.M.G., Bouzeghoub, M., Gardarin, G. (eds.) *EDBT 1996*. LNCS, vol. 1057, pp. 18–32. Springer, Heidelberg (1996)
10. Shafer, J.C., Agrawal, R., Mehta, M.: SPRINT: A scalable parallel classifier for data mining. In: Proc. 22nd International Conference Very Large Databases, pp. 544–555 (1996)
11. Alsabti, K., Ranka, S., Singh, V.: CLOUDS: A decision tree classifier for large datasets. In: Proc. Conference Knowledge Discovery and Data Mining (KDD 1998), pp. 2–8 (1998)
12. Gehrke, J., Ramakrishnan, R., Ganti, V.: Rainforest - a framework for fast decision tree classification of large datasets. In: Proc. of VLDB Conference, New York, pp. 416–427 (1998)
13. Gehrke, J., Ganti, V., Ramakrishnan, R., Loh, W.: BOAT - optimistic decision tree construction. In: Proc. of the ACM SIGMOD Conference on Management of Data, pp. 169–180 (1999)
14. Yoon, H., Alsabti, K., Ranka, S.: Tree-based incremental classification for large datasets. Technical Report TR-99-013, CISE Department, University of Florida, Gainesville, FL. 32611 (1999)
15. UCI machine learning repository, University of California (2007), <http://www.ics.uci.edu/mllearn/MLRepository.html>
16. Adelman-McCarthy, J., Agueros, M.A., Allam, S.S.: Data Release 6, ApJS, 175 (in press, 2008)

The Use of Semi-parametric Methods for Feature Extraction in Mobile Cellular Networks

A.M. Kurien, B.J Van Wyk, Y. Hamam, and Jaco Jordaan

French South African Technical Institute in Electronics (F'Satie),
Tshwane University of Technology, P/Bag X680, Pretoria, South Africa 0001
{KurienAm, VanWykB, HamamY}@tut.ac.za, jacojordan@webmail.co.za

Abstract. By 2006, the number of mobile subscribers in Africa outnumbered that of fixed line subscribers with nearly 200 million mobile subscribers across the continent [1][2]. By the end of 2007, it was estimated that the number of mobile subscribers would exceed 278 million subscribers [2]. Mobile Telephony has been viewed as a critical enabling technology that is capable of boosting local economies across Africa due to the ease of roll out of wireless technologies in comparison to fixed line networks. With the boom in wireless networks across Africa, a growing demand to effectively predict the rate of growth in demand for capacity in various sectors of the network has risen with cellular network operators. This paper looks at using *Spectral Analysis* techniques for the extraction of features from cellular network traffic data that could be linked to subscriber behavior. This could then in turn be used to determine capacity requirements within the network.

Keywords: Cellular networks, cellular network traffic, spectral analysis, feature extraction.

1 Introduction

Wireless networks across Africa have witnessed an incredible boom with rates of growth far exceeding expectations. In spite of having 34 of the poorest countries in world according to the UN, mobile penetration rates stood at 22.0 subscribers per 100 inhabitants [2][3]. Statistics have shown that the mobile market in Africa has been the fastest growing market over the last five years [1]. Africa had achieved a mobile subscriber growth rate of 46.2% between 2001 and 2005. The boom in the telecommunications market in most African countries has been fueled by the ease of deployment and novel business models which are prevalent in these countries. Mobile networks are vital part of most sectors of society in Africa on which people depend on for a livelihood [3]. Due to the constant growth in demand for capacity in the network, a systematic approach to cellular network planning is crucial to maintain effective growth and returns on investments [4]. With the availability of relatively accurate data from a cellular network, network planning that takes into consideration tele-traffic issues is vital for the long-term characterization of subscriber behavior [6]. With the drastically varying socio-economic status of various sectors of a typical developing

country, traffic trends that are generated from various sectors of a network can vary tremendously. By determining various types of traffic classes that contribute to the traffic loads in a given network, the long term traffic trends can be predicted for the purposes of capacity planning. The building of an accurate classification mechanism that is able to categorize various sectors of a mobile subscriber market into traffic classes and then predict the long term traffic demand that would be generated by the identified traffic classes would be beneficial to a mobile network provider in their network planning strategy. In this study, a sample of daily traffic variations of a typical urban area in South Africa is considered as an input to a feature extraction method that makes use of a typical robust graph fitting method to generate linear and non-linear (*bias*) components.

2 Tele-Traffic Modeling

Traditional network planning strategies employ an analytic approach in which radio network requirements are determined in the initial stages, focusing on radio frequency modeling [5]. The basic idea behind tele-traffic modeling in cellular networks lies in the ability of the model to capture important statistical properties of related data of the underlying environment. As mobile communications have radically changed from a technological perspective as well as from a service usage perspective, the design criteria of next generation cellular networks have to be altered [4]. One of the drawbacks of commonly used network planning methods is their inability to address the economical aspects of system deployment [6] and the neglect of factors such as user behavior and demand distribution. One method of predicting traffic generated within a region is by determining the type of region that the prediction is required for and then interpolating measured values obtained from network elements [7]. By assuming that each ground pixel in a service area generates an amount of traffic based on the land use type, the traffic generated within a cell may be determined using a linear combination of land use distribution values [7]. Traditional *tele-traffic* modeling techniques are practical only in environments that are well defined in terms of demographics and geography. One way of modeling traffic would be propose a mobility model to evaluate radio performance by monitoring *Base Station Site* (BSS) measurements [8]. Using statistical modeling techniques, it is possible to derive distributions that fit the measured data. The *Least Square Method* (LS) is a common method of fitting data to model functions of different complexities. The LS method determines the minimum averaged square deviations between sample points and a predictive line that passes through mean values of given 'x' and 'y' values [9]. The use of the Kolmogorov Smirnov goodness of fit test could be used for the derivation of distributions to fit measured data as shown in [8]. A more robust method for model fitting could be based on spectral estimation techniques. These methods may be used to distinguish between spectral lines that are very closely spaced in frequency. The benefit of the method is the automatic separation of components in comparison to similar methods. The following section describes the use of spectral estimation technique for model fitting.

3 Model Fitting Using Semi-parametric Methods

Spectral estimation methods can be classified into parametric and non-parametric methods. A combination of the two methods may be referred to as a semi-parametric method that is based on the method presented in [10] and [11]. An overview of the method is described below.

3.1 Overview of the Method

A system of linear differential equations is used as a mathematical model to represent the distribution of traffic generated in various cells in a typical cellular network. According to this assumption, the output signal $y(k)$ from such a system that is uniformly sampled at sampling times of kT_s can be parameterized using the sum of a series of n exponential functions as

$$y(k) = \sum_{i=1}^n A_i \exp(j\varphi_i) \exp[(d_i + j\theta_i)k], \tag{1}$$

where T_s is a constant sampling interval, $\theta_i = 2\pi f_i T_s$, and $d_i = \delta_i T_s$. The model parameters include frequencies, f_i , damping factors, δ_i , amplitudes, A_i and phases φ_i , $i = 1, 2, \dots, n$. Since real signals are only considered here, the signal poles defined by $\exp(\delta_i + j2\pi f_i)$, $i = 1, 2, \dots, n$, appear in complex conjugate pairs. The signal (1) complies with the linear difference equation (*Auto Regressive (AR) model*) and may be represented as follows.

$$y(k) + \sum_{i=1}^n x_i y(k-i) = 0, \quad k = n+1, \dots, n+m, \tag{2}$$

where x_i are the AR model coefficients and $n+m$ is the total number of signal samples. The signal model (1) is extended when analyzing the recorded signals in practical situations. There are additional signal components in some cases which originate from non-linearity, trends or unobservable control inputs. The extension of the signal model (1) for cellular network traffic applications is given by

$$y(k) = s(k) + \Delta s(k), \tag{3}$$

where $s(k)$ is the measured signal sample and $\Delta s(k)$ is the residual between the measured signal and the linear signal in equation (1). The residual is modeled using

$$\Delta s(k) = E[\Delta s(k)] + \varepsilon(k), \tag{4}$$

where $E[\Delta s(k)]$ (*the expected value of $\Delta s(k)$*) is the bias component that represents the trend in the variation in the cellular network traffic system and $\varepsilon(k)$ is the stochastic noise component which is assumed to be independent and identically distributed Gaussian noise. Substituting (3) into (4), the following constraint is obtained.

$$s(k) + \Delta s(k) + \sum_{i=1}^n x_i [s(k-i) + \Delta s(k-i)] = 0, \quad (5)$$

where $k = n + 1, \dots, n+m$.

3.2 Matrix Formulation

There are two possible matrix formulations of (5). The first formulation is obtained directly using

$$\mathbf{Ax} + \mathbf{b} + \Delta \mathbf{Ax} + \Delta \mathbf{b} = \mathbf{0}. \quad (6)$$

All the residuals are grouped together in the vector $\Delta \mathbf{s}$ as follows.

$$\mathbf{Ax} + \mathbf{b} + \mathbf{D}(\mathbf{x})\Delta \mathbf{s} = \mathbf{0} \quad (7)$$

3.3 Optimization Problem

If the linear model of order n is known in advance (*size of the coefficient vector, \mathbf{x}*), the optimal estimate of the residual vector $\Delta \mathbf{s}$ can be obtained by minimizing the second norm of the noise component while satisfying the constraints given in (5). When n is not known, the optimization problem can be formulated as an equality constrained least squares problem of minimizing the second norm of the noise component plus a penalty term. This puts a limit on the size of vector \mathbf{x} , i.e.

$$\begin{aligned} & \min_{\mathbf{x}, \Delta \mathbf{s}} \left\{ \frac{1}{2} [\Delta \mathbf{s} - \mathbf{E}(\Delta \mathbf{s})]^T [\Delta \mathbf{s} - \mathbf{E}(\Delta \mathbf{s})] + \frac{\mu}{2} \mathbf{x}^T \mathbf{x} \right\} \\ & = \min_{\mathbf{x}, \Delta \mathbf{s}} \left\{ \frac{1}{2} \Delta \mathbf{s}^T \mathbf{W} \Delta \mathbf{s} + \frac{\mu}{2} \mathbf{x}^T \mathbf{x} \right\} \end{aligned}$$

subject to $\mathbf{Ax} + \mathbf{b} + \mathbf{D}(\mathbf{x})\Delta \mathbf{s} = \mathbf{0}, \quad (8)$

where $\mathbf{W} = (\mathbf{I} - \mathbf{S})^T (\mathbf{I} - \mathbf{S})$. \mathbf{I} is the identity matrix, \mathbf{S} is the *Local Polynomial Approximation* (LPA) smoothing matrix used to estimate $\mathbf{E}(\Delta \mathbf{s})$ as $\mathbf{S}\Delta \mathbf{s}$, and μ is a penalty factor [12]. In (8), the initial size n of the coefficient vector \mathbf{x} should be assumed greater than the actual value. The penalized least squares method shrinks the coefficients by imposing a penalty on their size. This method automatically selects the order n by shrinking or setting some coefficients to zero. The LPA smoothing matrix is constructed by locally fitting (*on a moving window of selected data samples*) a polynomial of any order. Usually, a linear polynomial is used to smooth residuals $\Delta \mathbf{s}$ (*filter out noise*), and to estimate the expectation $\mathbf{E}(\Delta \mathbf{s})$ in the optimization problem of (8) [13]. It should be highlighted that the estimates are linear in $\Delta \mathbf{s}$ (*the smoothing matrix \mathbf{S} does not involve $\Delta \mathbf{s}$*).

3.4 Iterative Solution

The constrained optimization problem (8) can be reformulated using Lagrange multipliers as shown in the following expression.

$$L(\Delta\mathbf{s}, \mathbf{x}, \lambda) = \frac{1}{2} \Delta\mathbf{s}^T \mathbf{W} \Delta\mathbf{s} + \frac{\mu}{2} \mathbf{x}^T \mathbf{x} + \lambda^T [\mathbf{A}\mathbf{x} + \mathbf{b} + \mathbf{D}(\mathbf{x})\Delta\mathbf{s}], \tag{9}$$

where λ is a vector of Lagrange multipliers. A necessary condition for a local minimum of (9) is that the estimate $(\Delta\hat{\mathbf{s}}, \hat{\mathbf{x}}, \hat{\lambda})$ is a stationary point of the Lagrangian function (9). The stationary point condition at $(\Delta\hat{\mathbf{s}}, \hat{\mathbf{x}}, \hat{\lambda})$ is obtained by setting the derivatives of Lagrangian in (9) to zero, i.e.

$$\begin{aligned} \nabla_{\Delta\mathbf{s}} L &= \mathbf{0} \rightarrow \mathbf{W}\Delta\hat{\mathbf{s}} + \mathbf{D}(\hat{\mathbf{x}})^T \hat{\lambda} = \mathbf{0} \\ \nabla_{\mathbf{x}} L &= \mathbf{0} \rightarrow \mu \hat{\mathbf{x}} + (\mathbf{A} + \Delta\hat{\mathbf{A}})^T \hat{\lambda} = \mathbf{0} \\ \nabla_{\lambda} L &= \mathbf{0} \rightarrow \mathbf{A}\hat{\mathbf{x}} + \mathbf{b} + \mathbf{D}(\hat{\mathbf{x}})\Delta\hat{\mathbf{s}} = \mathbf{0}. \end{aligned} \tag{10}$$

To find the stationary point $(\Delta\hat{\mathbf{s}}, \hat{\mathbf{x}}, \hat{\lambda})$, the condition in (10) is linearized around $(\Delta\mathbf{s}^{(k)}, \mathbf{x}^{(k)}, \lambda^{(k)})$. The resulting Newton’s method generates a sequence of approximations to the stationary point by solving the linear system in each set of iterations.

4 Cellular Network Traffic Feature Extraction

Cellular network traffic data from typical cellular network in an urban area in South Africa is used as an input signal to the above method. The traffic data consists of periodic daily traffic distributions measured during peak-to-peak hours. The measurements were taken over a two year period. The traffic distributions were provided as input to the semi-parametric method to extract the linear component and the residual between the measured signal and the linear signal. A set of sample sites were selected to determine the variations that would be experienced in traffic distributions in the area under study. In the study, two distinct groups of site locations were considered. The first was a set of sites that existed in typical business activity areas such as central business districts or commercial areas. The second set consisted of sites

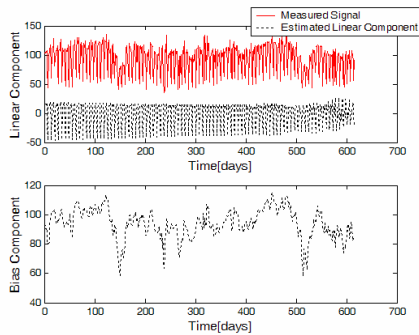


Fig. 1. Generated Linear and Bias (residual) components of measured signal (daily traffic) in a business area

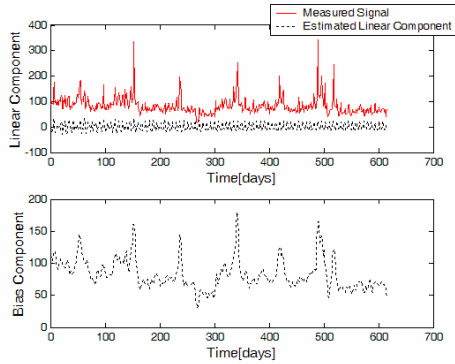


Fig. 2. Generated Linear and Bias (residual) components of measured signal (daily traffic) for suburban area

located in suburban areas in the area under study. The following figure shows the output generated for a site centered in a business area.

Similar outputs were generated for sites that were located in suburban areas. The output generated for one of the sites considered is shown in the following figure.

5 Interpretation of Results Obtained

One of the primary objectives of the study was to determine traffic variations that could lead to a rise in demand for capacity in the network, whether the traffic demands were seasonal (*occurring only for specific periods*), and the frequency of occurrence of the variations in demand for capacity. In the business type area, the traffic distribution of the sites considered yielded traffic dips that occurred at specific periods of the year. The dips in the traffic were cyclic and occurred on a yearly basis. This is accounted for due to the drop in business activity that experienced in these areas typically towards the end of the year. It was shown that dominant drops in the residual signal existed as illustrated in the following figure.

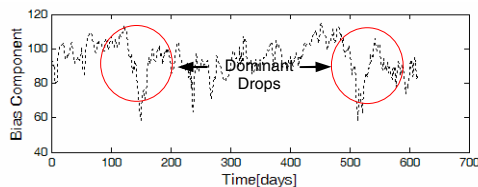


Fig. 3. Detection of the drop in traffic in business areas

The secondary peaks were indicative of secondary traffic variations experienced during the rest of the year (such as during vacation periods during the year) for which capacity provisioning would have to be made in the network to ensure optimum quality of service.

Similarly, for the suburban type area, the traffic distribution of the sites considered yielded in this case traffic peaks that occurred at specific periods of the year. The traffic peaks in the traffic were cyclic and occurred on a yearly basis, and some cases, more than once in a year. This is accounted for in the rise in traffic activity in the suburb area under consideration during typical holiday periods. This was experienced predominantly in the middle and end of the year, and in some cases, at quarterly intervals. In the case of suburban sites, the residual component showed dominant peaks in the distribution as illustrated in the following figure.

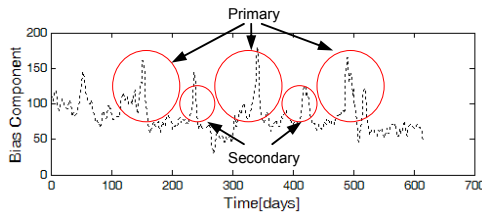


Fig. 4. Detection of traffic peaks in suburban areas

6 Conclusion

This study showed the use of *semi-parametric* methods for the purposes of model fitting of cellular network traffic. The purpose of the study was to determine if the method could be used for the extraction of features from typical cellular network traffic distributions. The extraction of linear and bias (residual) components of the signal was shown. For the two types of areas considered, the variation in traffic signals was clearly shown. It was shown that dominant dips in the signal occurred for sites in business areas whereas dominant peaks occurred for typical suburban areas.

The determination of capacity requirements for cellular network sites that experience traffic variations in the network is of great importance for a typical cellular network provider and the network planners and optimization teams. The increase in demand for capacity in the network needs to be determined to optimally provide for capacity enhancements in the network. This study showed a method of determining periodic variations in the traffic growth/slumps in the network. From this information, a provider is able to determine whether there is a need for capacity upgrades/downgrades at specific sites and whether a cyclic trend is experienced at specific sites. The benefit of the method is the ability of the method to detect dominant features that could be indicative of large shifts in the traffic behavior experienced at specific sites.

Acknowledgments. The authors would like to acknowledge M. Drewes and MTN South Africa for their valuable contribution to this work.

References

1. ICT in Africa: Forging a Competitive Edge, ITU Connect Africa (October 2007), http://www.itu.int/ITU-D/connet/africa/2007/media/kit/africa_ict.html
2. Telecommunication/ICT Markets and Trends in Africa 2007, Connect Africa Summit, Kigali, Rwanda (October 2007), <http://www.itu.int>
3. Mbarika, V.W.A., Mbarika, I.: Africa Calling [Africa Wireless Connection]. *IEEE Spectrum* 43(5), 56–60 (2006)
4. Tutschku, K.: Demand-Based Radio Network Planning of Cellular Mobile Communication Systems. In: *IEEE Conference on Computer Communication (INFOCOM)*, April 1998, (1), pp. 1054–1061 (1998)
5. Tutschku, K., Leibnitz, K., Tran-Gia, P.: ICEPT- An Integrated Cellular Network Planning Tool. In: *Vehicular Technology Conference (VTC)*, Phoenix, USA, May 1997, pp. 765–769 (1997)
6. Tutschku, K., Tran-Gia, P.: Spatial Traffic Estimation and Characterization for Mobile Communication Network Design. *IEEE Journal on Selected Areas in Communications* 16(5), 804–811 (1998)
7. Gawrysiak, P., Okoniewski, M.: Applying Data Mining Methods for Cellular Radio Network Planning. In: *Proceedings of the International Intelligent Information Systems Conference, Advances in Soft Computing Journal*. Springer, Heidelberg (2000)
8. Khedher, H., Valois, F., Tabbane, S.: Traffic Characterization for Mobile Networks. In: *56th IEEE Vehicular Technology Conference (VTC)*, September 2002, vol. 3, pp. 1485–1489 (2002)
9. Camilleri, M.: Forecasting Using Non-Linear Techniques in Time Series Analysis: An Overview of Techniques and Main Issues. In: *Computer Science Annual Research Workshop (CSAW 2004)*, September 2004, pp. 19–28 (2004)
10. Zivanovic, R.: Analysis of Recorded Transients on 765kV Lines with Shunt Reactors. In: *Proc. Power Tech 2005 Conference*, St. Petersburg, Russia (2005)
11. Zivanovic, R., Schegner, P., Seifert, O., Pilz, G.: Identification of the Resonant-Grounded System Parameters by Evaluating Fault Measurement Records. *IEEE Transactions on Power Delivery* 19(3), 1085–1090 (2004)
12. Draper, N., Smith, H.: *Applied Regression Analysis*, 2nd edn. John Wiley & Sons, Chichester (1981)
13. Jordaan, J.A., Zivanovic, R.: Time-varying Phasor Estimation in Power Systems by Using a Non-quadratic Criterium. *Transactions of the South African Institute of Electrical Engineers (SAIEE)*, 95(1), 35–41 (March 2004) ERRATA: 94(3), 171–172 (September 2004)

Personalized Document Summarization Using Non-negative Semantic Feature and Non-negative Semantic Variable

Sun Park

Department of Computer Engineering, Honam University, Gwangju, Korea
sunpark@honam.ac.kr

Abstract. Recently, the necessity of personalized document summarization reflecting user interest from search results is increased. This paper proposes a personalized document summarization method using non-negative semantic feature (NSF) and non-negative semantic variable (NSV) to extract sentences relevant to a user interesting. The proposed method uses NSV to summarize generic summary so that it can extract sentences covering the major topics of the document with respect to user interesting. Besides, it can improve the quality of personalized summaries because the inherent semantics of the documents are well reflected by using NSF and the sentences most relevant to the given query are extracted efficiently by using NSV. The experimental results demonstrate that the proposed method achieves better performance the other methods.

1 Introduction

With the fast growth of the Internet access by personal user, it has increased the necessity of the personalized information seeking and personalized summaries. The automatic summarization is the process of reducing the sizes of documents while maintaining their basic outlines. That is, it should distill the most important information from the document. It can be either generic summaries or query based summaries. A generic summary distills an overall sense of the documents' contents whereas a query based summary only distills the contents of the document relevant to the user's query [5].

If the summary is personalized according to user interests, the user can save time not only in deciding whether it is interesting or not, but also in finding the information without having to read the full text. The personalized summarization is the process of summarization that preserves the specific information that is relevant for a given user profile rather than information that truly summarizes the content of the search results [1]. To build a personalized or user-adapted summary a representation of the interests of the corresponding user is studied [1, 9, 11, 12, 13]. Diaz and Gervas proposed personalized summarization using combination of generic method by using the information of position and personalized method by using similarity between sentences and user interesting [1]. However, this method may produce poor personalized summarization in the case that the information of position and similarity does not reflect the user interesting. Park also proposed personalized summarization using combination of generic method

by using Relevance Measure (RM) and query-based summarization by using non-negative matrix factorization (NMF) [9]. However, this method may select the meaningless sentences in the case that user interesting does not reflect the generic summaries, since the RM use a simple similarity between sentence and whole document.

In this paper, we propose a method using non-negative semantic features and non-negative semantic variables by NMF to summarize a personalized summarization with regard to a given query. The NMF can find a parts representaion of the data because non-negative constraints of the NMF are compatible with the intuitive notions of combining parts to form a whole, which is how the NMF learns a parts-based representation. Also it can represent a large volume of information efficiently because of the sparsely distributed representation of NMF [3, 4, 14]. The non-negative semantic feature matrix (NSFM) and the non-negative semantic variable matrix (NSVM) are calculated from NMF.

The propose method has the following advantages: First, it can select sentences covering the major topics of the document with respect to user interesting by using NSV. Second, it can improve the quality of document summarization since extracting sentences to reflect the inherent semantics of a document by using NSFM and NSVM. Third, it can select sentences that are highly relevant to a user interesting because it can chooses the sentences related to the user’s query relevant semantic features that well represent the structure of a document.

The rest of the paper is organized as follows: Section 2, we describe the NMF algorithm in detail and In Section 3, the personalized document summarization method is introduced. In Section 4 describe Related Works. In Section 5 shows the evaluation and experimental results. Finally, we conclude in Section 6.

2 Non-negative Matrix Factorization

In this paper, we define the matrix notation as follows: Let X_{*j} be j 'th column vector of matrix X , X_{i*} be i 'th row vector, and X_{ij} be the element of i 'th row and j 'th column.

Non-negative matrix factorization (NMF) is to decompose a given $m \times n$ matrix A into a non-negative semantic feature matrix W and a non-negative semantic variable matrix H as shown in Equation (1).

$$A \approx WH \tag{1}$$

where W is a $m \times r$ non-negative matrix and H is a $r \times n$ non-negative matrix. Usually r is chosen to be smaller than m or n , so that the total sizes of W and H are smaller than that of the original matrix A .

We use the objective function that minimizes the Euclidean distance between each column of A and it approximation $\tilde{A} = WH$, which was proposed in Lee and Seung [3, 4]. As an objective function, the Frobenius norm is used [14]:

$$\Theta_E(W, H) \equiv \|A - WH\|_F^2 \equiv \sum_{j=1}^m \sum_{i=1}^n \left(X_{ji} - \sum_{l=1}^r W_{jl} H_{li} \right)^2 \tag{2}$$

We keep updating W and H until $\Theta_E(W, H)$ converges under the predefined threshold or exceeds the number of repetition. The update rules are as follows:

$$H_{\alpha\mu} \leftarrow H_{\alpha\mu} \frac{(W^T A)_{\alpha\mu}}{(W^T WH)_{\alpha\mu}}, W_{i\alpha} \leftarrow W_{i\alpha} \frac{(AH^T)_{i\alpha}}{(WHH^T)_{i\alpha}} \tag{3}$$

3 Personalized Document Summarization Using NSF and NSV

In this paper, we propose a personalized summarization method using NMF and NSV. The proposed method consists of the preprocessing phase, the generic summarization phase, the personalized summarization phase, and combination of generic and personalized summarization phase.

3.1 Preprocessing Phase

In the preprocessing phase, after given documents are decomposed into individual sentences, we remove stop-words and perform words stemming. Then we construct the weighted term-frequency vector for each sentence in documents using Equation (4) [10]. Let A be $m \times n$ matrix, where m is the number of terms and n is the number of sentences in the whole documents. Let element A_{ji} be the weighted term-frequency of term j in sentence i .

$$A_{ji} = L_{ji} \cdot G(j) \tag{4}$$

where L_{ji} is the local weight(term frequency) for term j in the sentence i , and $G(j)$ is the global weight(inverse document frequency) for term j in the whole documents [10]. That is,

$$G(j) = \log(N/N(j)) \tag{5}$$

where N is the total number of sentences in the whole documents, and $N(j)$ is the number of sentences that contain term j .

3.2 Generic Summarization Phase

We modify our previous generic summarization method using non-negative semantic variable by NMF [8]. We define the *semantic weight of a sentence* $weight(\)$ as follows:

$$weight(H_{i*}) = \sum_{q=1}^n H_{iq} \tag{6}$$

The weight (H_{i*}) means the relative relevance of i 'th semantic feature (W_{*i}) among all semantic features. The generic relevance of a sentence means how much the sentence reflects major topics which are represented as semantic features.

We compute the relevance score of each selected sentence with a user interesting by Equation (7). The relevance score means how much the selected sentence reflects user interesting which are represented as the semantic weight of a sentence.

$$r_i = weight(H_{i*}) \times \overset{\rightarrow}{sim}(q, A_{*i}) \times o_k \tag{7}$$

where r_i is a relevance score of i 'th sentence, $sim()$ is a cosine similarity function, \vec{q} is a query for user interesting, A_{*i} is a i 'th sentence, o_k is a order score.

The cosine similarity function between the vector of a 'th sentence A_{*a} and the vector of b 'th sentence A_{*b} is computed as follows [10].

$$sim(A_{*a}, A_{*b}) = \frac{A_{*a} \cdot A_{*b}}{|A_{*a}| \times |A_{*b}|} = \frac{\sum_{j=1}^m A_{ja} \times A_{jb}}{\sqrt{\sum_{j=1}^m A_{ja}^2} \times \sqrt{\sum_{j=1}^m A_{jb}^2}} \tag{8}$$

We define the order score o_k as Equation (9).

$$o_k = 1.00 - 2 \times (k - 1) \times 0.01 \tag{9}$$

where k is a number of the selected sentences for generic document summary. The order score o_k reflects the weight of rank of the selected sentence to the relevance score.

The proposed algorithm for generic document summarization is as follows:

1. Decompose the document D into individual sentences, and let k be the number of sentences for generic document summarization.
2. Perform the stopwords removal and words stemming operations.
3. Construct the weighted terms by sentences matrix A using Equation (4).
4. Perform the NMF on the matrix A to obtain the matrix H using Equation (3)
5. for each sentence j
 calculate the *semantic weight of sentence j* $weight(H_{j*})$.
6. Select k sentences with highest semantic weight values, and add it to the candidate sentence set.
7. Compute the relevance score for each selected sentences using Equation (7)

3.3 Personalized Document Summarization Phase

We modify our previous query based document summarization method [6] using NMF for personalized summarization. The personalized summarization phase is described as follows.

To evaluate the degree of similarity of the semantic feature vector W_{*l} with regard to the query \vec{q} as the correlation between the vector W_{*l} and \vec{q} used Equation (8).

To select those sentences of a document that is most relevant to a given query, personalized score e_i , is defined as follows.

$$e_i = o_k \times sim(W_{*l}, \vec{q}) \tag{10}$$

where e_i is a personalized score of i 'th sentence, o_k is order score, k is the number of extract sentences.

We use the following personalized summarization algorithm:

1. Decompose the document D into individual sentences, and let k be the number of sentences for the selected sentences.
2. Perform the preprocessing phase.

3. Construct the weighted terms by sentences matrix A using Equation (4).
4. Perform the NMF on the matrix A to obtain the matrix W and the matrix H using Equation (3).
5. Select a column vector W_{*p} of matrix W whose similarity to the query is the largest using Equation (8).
6. Select the sentence corresponding to the largest index value of the row vector H_{p*} , and include it in the candidate sentences set.
7. Compute the personalized score e_i using Equation (10).
8. If the number of selected sentences reaches the predefined number k , then stop the algorithm. Otherwise go to step 5 to find the next most similar column vector excluding W_{*p} .

In step 5, the fact that the similarity between W_{*p} and the query is largest means the p 'th semantic feature vector W_{*p} is the most relevant feature to the query. In step 6, it selects the sentence that has the largest weight for the most relevant semantic feature.

3.4 Combination of Generic and Personalized Document Summarization Phase

The combination of generic and personalized summarization phase extracts top k ranked sentences from the candidate sentences set for automatic personalized summary. This phase is described as follows: We normalize the relevance scores and the personalized scores. We then calculate the ranking score of the candidate sentences by using Equation (11).

$$rs_i = r_i + e_i \quad (11)$$

where rs_i is a ranking score of i 'th sentence.

4 Related Works

The previous studies for personalized summarization are as follows: Tombros and Sanderson proposes a query biased summaries. Their method generates user adapted summaries by combining title score, heading score, term occurrence information, and query score [12].

Sanderson proposes a accurate user directed summarization using best passage operator and query expansion from Local Context Analysis (LCA). His method selects the passage most relevant to the query, previously expanded with the words that occur most frequently in the context in which the words of the query appear in the first document retrieved [11].

Varadarajan and Hristidis proposes query specific document summarization method by identifying the most query relevant fragments and combining them using the semantic associations within the document [13].

Diaz and Gervas propose automatic personalized summarization using combination of generic and personalized methods. Their generic summarization methods combine position method with thematic word method. Their personalized method is to select those sentences of a document that are most relevant to a given user model [1].

Park proposed a automatic personalized summarization using NMF and Relevance Measure [9]. This method use combination of generic by Relevance Measure and personalized summarization method by NMF to extract the important sentences.

Park et al. proposed a query based summarization method using NMF [6]. This method extracts sentences using the similarity between a query and semantic features extracted by NMF. Park et al. also proposed a multi-document summarization method based on clustering using NMF [7]. This method clusters the sentences and extracts sentences using the cosine similarity measure between a topic and semantic features. This method improves the quality of summaries and avoids the topic to be deflected in the sentence structure by clustering sentences and removing noise. Park proposed a generic summarization method using non-negative semantic variable by NMF [8]. This method extracts sentence to reflect major topics in document which are represented as semantic features.

5 Experimental Results

As an experimental data, we used Yahoo-Korea News¹. We gave a query to retrieve news documents from Yahoo-Korea News. Three independent evaluators were employed to manually create summarization on the 200 document from the retrieved Yahoo Korea news documents with respect to 10 queries. Each document in Yahoo Korea news has an average of 9.08 sentences selected by evaluators. Table 1 provides the particulars of the evaluation data set. The retrieved news documents are preprocessed using HAM (Hangul Analysis Module) which is a Korean language analysis tool based on Morpheme analyzer [2].

Table 1. Particulars of the Evaluation Data Corpus

Document attributes	Yahoo-Korea News
Number of docs	200
Number of docs with more than 10 sentences	136
Avg sentences / doc	10.1
Min sentences / doc	9
Max sentences / doc	86

In this paper, we used the recall (R), precision (P), and F -measure to evaluate the performance of the proposed method. Let S_{man} , S_{sum} be the set of sentences selected by the human evaluators, and the summarizer, respectively. The standard definitions of recall (R), precision (P), and F -measure are defined as follows [10]:

$$R = \frac{|S_{man} \cap S_{sum}|}{|S_{sum}|}, P = \frac{|S_{man} \cap S_{sum}|}{|S_{man}|}, F = \frac{2RP}{R+P} \quad (12)$$

We evaluated 4 different summarization methods such as the UPS, the QNMF, the PNMFR, and the PSFV. The UPS denotes Diaz's method [1] using user-model based

¹ <http://kr.news.yahoo.com> (2008)

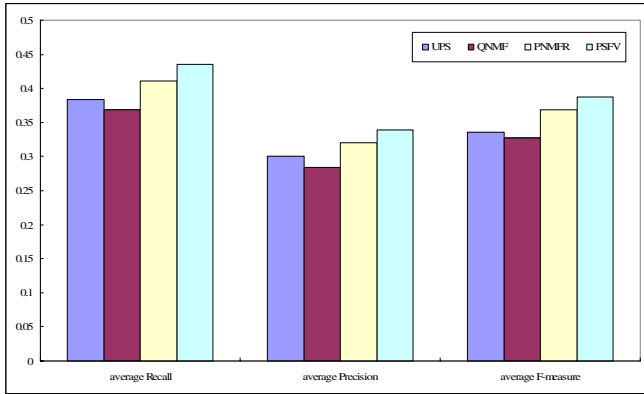


Fig 1. Evaluation Results using Yahoo-Korea News

personalized summarization. The QNMF denotes our previous method using query based document summary by QNMF [6]. The PNMFR also denotes our previous method using NMF and Relevance Measure for personalized summary [9]. The PSFV denotes the proposed method.

The evaluation results are shown in Figure 1. The average recall of PSFV is approximately 15.59% higher than that of QNMF, 11.93% higher than that of UPS, and 5.73% higher than that of PNMFR. The average precision of PSFV is approximately 16.22% higher than that of QNMF, 11.21% higher than that of UPS, and 5.31% higher than that of PNMFR. The average F-measure of PSFV is approximately 15.50% higher than that of QNMF, 13.18% higher than that of UPS, and 4.91% higher than that of PNMFR.

The result shows that recall, precision, and F-measure of UPS are better than those of the QNMF because the UPS influences generic summary to user's interesting. The result shows that recall, precision, and F-measure of PNMFR are better than those of the UPS because the PNMFR generates more meaningful summary by reflecting the inherent semantics of document with respect to generic summary. The PSFV shows best performance. The proposed method generates meaningful personalized summary by means of the semantic feature and semantic variable reflecting the inherent structure in document for user interesting.

6 Conclusion

A personalized summary adapt to user to correctly identify whether a document is really interesting for him without having to read the whole document. In this paper, we use semantic variable and cosine similarity to summarize the generic summary. Also we apply NMF to reflect the inherent semantics of documents for personalized summary. We use combination of generic and personalized method to extract the meaningful sentences. The proposed method can select sentences covering the major topics of the document with respect to user interesting. Besides, it can improve the quality of document summarization since extracting sentences to reflect the inherent

semantics of a document with respect to a given query. Experimental results show that the proposed method outperforms the 3 different summarization methods.

In the near future, we plan to enhance the personalized summarization method by using a variety of weighting terms. We anticipate that it can improve the accuracy of automatic personalized summarization.

References

- [1] Diaz, A., Gervas, P.: User-model based personalized summarization. *Information Processing and Management* 43, 1715–1734 (2007)
- [2] Kang, S.S.: *Information Retrieval and Morpheme Analysis*. HongReung Science Publishing Company (2002)
- [3] Lee, D.D., Seung, H.S.: Learning the parts of objects by non-negative matrix factorization. *Nature* 401, 788–791 (1999)
- [4] Lee, D.D., Seung, H.S.: Algorithms for non-negative matrix factorization. *Advances in Neural Information Processing Systems* 13, 556–562 (2000)
- [5] Mani, I.: *Automatic Summarization*. John Benjamins Publishing Company, Amsterdam (2001)
- [6] Park, S., Lee, J.H., Ahn, C.M., Hong, J.S., Chun, S.J.: Query Based Summarization using Non-negative Matrix Factorization. In: *Proceeding of the International Conference on Knowledge-Based & Intelligent Information & Engineering Systems*, pp. 84–87 (2006)
- [7] Park, S., Lee, J.H.: Topic-based Multi-document Summarization Using Non-negative Matrix Factorization and K-means. *Journal of KIISE: Software and Applications* 35(4), 255–264 (2008)
- [8] Park, S.: Generic Summarization using Non-negative Semantic Variable. In: *Proceeding of the International Conference on Intelligent Computing*, pp. 1052–1058 (2008)
- [9] Park, S.: Automatic Personalized Summarization using Non-negative Matrix Factorization and Relevance Measure. In: *Proceeding of the IEEE International Workshop on Semantic Computing and Applications*, pp. 72–76 (2008)
- [10] Ricardo, B.Y., Berthier, R.N.: *Modern Information Retrieval*. ACM Press, New York (1999)
- [11] Sanderson, M.: Accurate user directed summarization from existing tools. In: *Proceeding of the international conference on information and knowledge management*, pp. 45–51 (1998)
- [12] Tombros, A., Sanderson, M.: Advantages of Query Biased Summaries in Information Retrieval. In: *Proceeding of ACM SIGIR*, pp. 2–10 (1998)
- [13] Varadarajan, R., Hristidis, V.: A System for Query-Specific Document Summarization. In: *Proceeding of the CIKM*, pp. 622–631 (2006)
- [14] Wild, S., Curry, J., Dougherty, A.: Motivating Non-Negative Matrix Factorizations. In: *Proceeding of SIAM ALA* (2003)

Cooperative E-Organizations for Distributed Bioinformatics Experiments

Andrea Bosin¹, Nicoletta Dessì¹, Mariagrazia Fugini², and Barbara Pes¹

¹ Università degli Studi di Cagliari, Dipartimento di Matematica e Informatica,
Via Ospedale 72, 09124 Cagliari
andrea.bosin@dsf.unica.it, dessi@unica.it, pes@unica.it

² Politecnico di Milano, Dipartimento di Elettronica e Informazione,
Piazza da Vinci 32, I-20133 Milano
fugini@elet.polimi.it

Abstract. Large-scale collaboration is a key success factor in today scientific experiments, usually involving a variety of digital resources, while Cooperative Information Systems (CISs) represent a feasible solution for sharing distributed information sources and activities. On this premise, the aim of this paper is to provide a paradigm for modeling scientific experiments as distributed processes that a group of scientists may go through on a network of cooperative e-nodes interacting with one another in order to offer or to ask for services. By discussing a bioinformatics case study, the paper details how the problem solving strategy related to a scientific experiment can be expressed by a workflow of single cooperating activities whose implementation is carried out on a prototypical service-based scientific environment.

Keywords: E-organizations, Cooperative Information Systems, Bioinformatics.

1 Introduction

Scientific researches vary greatly in their complexity, function and application and embrace information technology at all levels. Among the most common ICT technologies supporting scientific laboratories there are toolkits specifically aimed at supporting experiments and general purpose software tools that are still essential in enabling them (e.g., graphical tools, mathematical tools, data mining tools). This combination of computational resources can be especially effective in conjunction with the use of more widely deployed infrastructures such as Web Services or Grid computing.

As many present scientific experiments start to generate lots of data that are often scattered over many sites, a networked computer is no longer a technical support, but becomes an integrated part of the experiment. On these premises, the concept of “what an experiment is” moves from the idea of a local laboratory activity towards a computer and network supported *cooperative application*. For such applications, the concept of Cooperative Information System (CIS) [1] offers feasible solutions for interconnection, integration and large information sources sharing during experiment planning and execution.

However, the definition of a *scientific CIS* poses hard technical challenges deriving from the need of data access and integration, as well as from the scale, heterogeneity, distribution and dynamic variation of information. Specifically, the design of a *scientific CIS* involves at least two basic issues:

- the definition of a conceptual collaboration framework stating the virtual environment where a cooperative experiment is carried out;
- the design of a distributed architecture for executing cooperative scientific experiments based on a platform for integrated use of heterogeneous applications and software tools.

According to the conceptual framework presented in [2], this paper describes a service-based approach aimed at defining a *scientific CIS* for scientific experiments. The proposed paradigm models scientific experiments as “ad hoc” business processes executed by groups of scientists on a network of cooperative e-nodes interacting to offer, or to ask for, services. Each e-node has a local configuration and a set of shared resources, while services correspond to different scientific functionalities across research domains, and encapsulate computing and simulation capabilities.

The problem solving strategy related to a scientific experiment is modeled as a workflow of single cooperating activities, each performed by a set of e-services in a global experimental environment. The physical distribution of resources and access problems are masked via the service interfaces.

This paper focuses on experiments in the area of bioinformatics, although the proposed experimental space can be extended to various application fields such as mathematics, physics and computational science.

The paper is organized as follows. Section 2 reviews related work on cooperative systems and e-services. Section 3 describes the proposed scientific CIS. As a case study, Section 4 focuses on micro-array data classification experiments and presents a prototypical application environment based on Web Services. Finally, Section 5 gives the conclusions and future work.

2 Related Work

Various studies in the literature have stressed that a key success factor to promote research intensive products is the vision of a large scale scientific exploration carried out in a networked cooperative environment in the style of Cooperative Information Systems [1]. The focus is on high performance computing infrastructures, e.g. of grid type [3], supporting flexible collaboration and computation on a global scale [4]. The availability of such open virtual cooperative environments should lower barriers among researchers taking advantage of individual innovation and allowing the development of collaborative scientific experiments [5].

By applying Web Services [6] and grid computing [3], an experiment or a simulation can be executed in a cooperative way on various computational nodes of a network, allowing resource exchange among researchers. Moreover, cooperation among scientists involves the integration of a variety of information and data sources, the interaction with physical devices, as well as the use of existing software systems. This scenario is similar to that of enterprise environments, whose progress requires large-scale cooperative processes and efficient access to very large data collections and

computing resources [7][8]. Hence, existing enterprise models can be useful for designing cooperative scientific experiments as *scientific CIS* [2].

3 The Cooperative Service-Based Framework

We aim to define a collaborative environment where, analogously to a manager in a Virtual Enterprise, a *Chief Scientist* plans, designs and coordinates a scientific experiment in a temporary and dynamic virtual organization. Such organization is enabled by geographically distributed and networked laboratories each devoted to information processing.

Specifically, a scientific experiment is regarded as a process whose tasks can be decomposed and made executable as granular services individually. The decomposition is formalized by a workflow designed by the Chief Scientist who is in charge of coaching partners through the entire experiment life-cycle, from the definition of the experiment workflow to the results validation.

While many networked resources (services, applications, databases, and so on) can be associated to a single functional task, the design of the experiment workflow is based on a structured and complete vision of the experimental processes and of the available resources that we model in a single framework inspired by the SOA paradigm [9]. As shown in Fig. 1, the framework has a layered structure involving four levels of abstraction.

At the *process layer*, the experiment is formalized by an abstract workflow out of the available scientific services. The workflow includes points of access to remote information sources and points of cooperation with other e-nodes as well as possible destinations of processed information. At this level, the scientist uses his/her knowledge of the scientific process to connect the dots between the experimental strategy and its execution that occurs in the lower layers.

The *service layer* models granular tasks performed by each e-node or other local control activities. Services organize their activity on the basis of both local and network information sources and are related to the particular scientific context through the workflow describing the experiment plan.

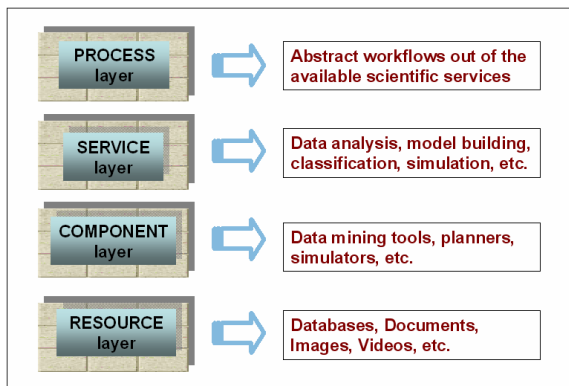


Fig. 1. The cooperative service-based framework for scientific experiments

The *component layer* expresses the software components invoked at the service layer, while the *resource layer* collects physical resources related to the activities assigned to each e-node. Their granularity and type can vary from the file, to the database or data warehouse, to the physical devices used in the experiment. Portions of local information sources can be declared as “public” by the e-node and hence become part of the network (i.e., remotely accessible).

The illustrated framework defines the service-based organization of processes and resources in carrying out collaborative experiments where the physical distribution of resources and access problems are masked via the service interface. As new information sources and processing techniques become available, they are simply represented as new services at suitable levels thus ensuring the environment scalability.

4 A Case Study

To illustrate how a scientific experiment can be formalized and executed in the outlined framework, let us consider a case study from the bioinformatics field. Specifically, we focus on processing micro-array data since they exemplify a situation that will be increasingly common in applications of machine learning to molecular biology. Micro-array technology [10] enables to put the probes for the genes of an entire genome onto a chip, such that each data point provided by an experimenter lies in the high-dimensional space defined by the size of the genome under investigation. Related data are characterised by many variables, involving up to thousands of genes, but only a handful of observations, the biological samples. As such, micro-array data are prime examples of extremely small sample-size but high-dimensional data.

Any micro-array experiment involves a number of activities (i.e., image processing, feature selection, clustering of genes with similar expression profiles, etc.). We focus on the classification experiments aiming at identifying genes whose expression patterns have meaningful biological relationships to a specific physio-pathological condition [11], [12], [13].

4.1 Characterizing Micro-array Classification Services

A collaborative environment for solving micro-array data classification problems should include the following minimal set of services.

Data Extraction Services. Online in the Web, a wide variety of useful scientific and in particular bio data are available for classification experiments. Data can be downloaded from many organizations, each identified through an URL address, and usually differ by their format (i.e. RES, GCT, CLS etc). Data Extraction Services deal with pre-processing of these data in order to map different dataset categories into a format suitable for the classification process.

Experiment Services. In terms of experiment design, the classification process can be partitioned into functional sub-modules, that we call *experiment services*, where

single computing functions are isolated and exposed as specific services to facilitate modularity and re-use. Typical experiment services are data mining algorithms and feature selection procedures both available as open-source or proprietary tools.

Visualization Services. Visualization Services deal with the proper visualization of the experiment results in graphs and/or textual descriptions accessible from any web browser.

4.2 Running Micro-array Classification Experiments

According to the proposed approach, the experiment is carried out on a network of cooperative e-nodes interacting with one another in order to offer or to ask for services. Based on J2EE [14] and Oracle [15] platforms, we implemented a prototypical e-node that we call *local pool* since it can be conceived as grouping a set of different local digital resources. The e-node makes services available on a data mining tool operating on a relational DBMS.

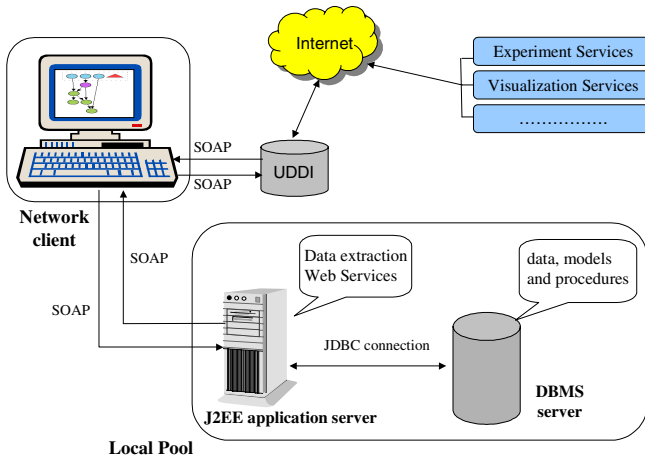


Fig. 2. The system deployment

As Fig. 2 shows, a network client allows the scientist to define the workflow that expresses the experiment plan. A data extraction service is offered as a local service dealing with pre-processing of data downloaded from public repositories in order to allow their storage into a DBMS according to a suitable format. Data extraction is carried out by a Java module exposed as a Web Service in order to be adopted and invoked by other experiments. Experiment and visualization services available on the Internet can be selected through an UDDI registry and included into the experiment workflow.

The Chief Scientist creates the workflow model that characterizes the distributed experiment in terms of selection and choreography of resources available both on the

Web and at the local site. The actor who performs the workflow design is the domain expert, often not very familiar with low level technical problems. To face this issue, our implementation is based on Taverna [16] that has been designed in order to assist a workflow definition and execution and to analyze its outputs. Whereas other tools stress technical requirements, Taverna is mainly concerned with planning activities. Through searching, selecting and linking Web Services, Taverna allows the graphical definition of the experiment workflow and makes it possible to discover and select the necessary resources.

4.3 Planning Micro-array Classification Experiments

The experiment planning involves two phases. First, as modeled at the process layer of the proposed framework, the scientist locates the digital resources needed to perform the experiment and defines the experiment workflow by using a graph structure. The workflow details the experiment plan according to available resources (accessed as Web Services or grid services) as well input and output data.

During the second phase, the workflow is transformed into an operational schema that provides a more formal specification of services that the framework models at the second layer. This schema is expressed in the ScufI format [17] and saved as an XML file. Then, the workflow and the associated experiment get executed in a cooperative way on the network and results are stored into suitable locations (e.g., into a database or in specified network storage areas) as specified by the workflow.

As an example of experiment planning, we consider a micro-array data classification problem related to acute myeloid and lymphoblastic leukemia [18] and to acute lymphoblastic leukemia [19] datasets, both investigated in [11][12][13].

The network resources available for this experiment are a set of Web Services [20] that provide the following classes of experiment services:

- *Feature ranking/selection services*: extract a (possibly small) subset of attributes from the original dataset. Significant attributes are kept, while irrelevant attributes are removed.
- *Filtering services*: reduce the size of the original dataset by keeping only a subset of all the attributes (e.g., those selected by feature selection), and remove all the others.
- *Classification services*: build a classification model (classifier) for the given dataset, using classification algorithms such as Bayesian Networks, Support Vector Machines, k-Nearest-Neighbor, etc.
- *Model testing services*: test the classification model on an independent set of data and measure the number of correctly classified vs. misclassified instances.
- *Model application services*: apply the classification model to a set of unlabeled data, and make a prediction of the class (label) of each instance.

Fig. 3 shows an example of workflow generated with Taverna for planning and executing the micro-array classification experiments described above. The workflow components are described in the following. The arrows represent the flow of data among available services.

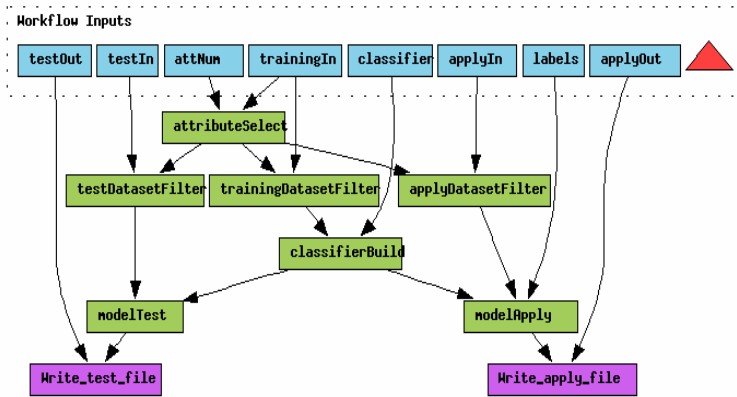


Fig. 3. The experiment workflow

Workflow inputs:

- *trainingIn*, *testIn*, *applyIn*: input files for classification, model testing and model application services
- *attNum*: number of attributes used in feature selection
- *classifier*: the specific classification service used to build the classification model
- *labels*: list of labels (classes) used in making predictions
- *testOut*, *applyOut*: output files for model testing and model application services

Workflow services:

- *AttributeSelect*: performs a feature selection on a dataset and outputs a list of features (Web Service)
- *DatasetFilter*: performs filtering on a dataset and output the reduced dataset (Web Service)
- *ClassifierBuild*: builds, encodes and outputs the classification model (Web Service)
- *ModelTest*: tests/validates the classification model and outputs results (Web Service)
- *ModelApply*: applies model and outputs results (Web Service)
- *Write_Text_File*: writes data into a file (I/O tool)

5 Conclusions

We have illustrated how a service-based approach can be applied for modeling scientific cooperative experiments. Even if focused on bioinformatics, the proposed approach is quite general and flexible, being adaptable to different scientific contexts. What remains open, however, are many questions about the realization of a *scientific CIS*. The implementation effort described here involves just a prototypical environment based upon Web Services technology and applied to data mining experiments. In future research works, we are going to implement the proposed approach to experiments that require a parameterized execution of a large number of jobs: each is executed in a dynamic environment where resources are unknown a priori and may

need agile adaptation to changes. Moreover, we plan to study how to monitor and control workflow execution including hierarchical execution with sub-workflows created and destroyed when necessary.

References

- [1] Meersman, R., Tari, Z. (eds.): Proceedings of the 15th International Conference on Cooperative Information Systems, Vilamoura, Portugal, November 28-30 (2007)
- [2] Bosin, A., Dessì, N., Fugini, M.G., Liberati, D., Pes, B.: Applying Enterprise Models to Design Cooperative Scientific Environments. In: Bussler, C.J., Haller, A. (eds.) BPM 2005. LNCS, vol. 3812, pp. 281–292. Springer, Heidelberg (2006)
- [3] Berman, F., Fox, G., Hey, T. (eds.): Grid Computing: Making the Global Infrastructure a Reality. John Wiley and Sons, Inc., New York (2003)
- [4] De Roure, D., Gil, Y., Hendler, J.A. (eds.): IEEE Intelligent Systems, Special Issue on E-Science, vol. 19(1) (2004)
- [5] <http://www.mygrid.org.uk/>
- [6] Alonso, G., Casati, F., Kuno, H., Machiraju, V.: Web services - Concepts, architectures, and applications. Springer, Heidelberg (2004)
- [7] Pollock, J.T., Hodgson, R.: Adaptive Information: Improving Business Through Semantic Interoperability, Grid Computing, and Enterprise Integration. Wiley Series in Systems Engineering and Management. Wiley-Interscience, Chichester (2004)
- [8] Travica, B.: Virtual organization and electronic commerce. ACM SIGMIS Database 36(3) (2005)
- [9] Singh, M., Huhns, M.: Service-oriented computing: Semantics, processes, agents. Wiley, Chichester (2005)
- [10] Hardimann, G.: Microarray methods and applications: Nuts & bolts. DNA Press (2003)
- [11] Bosin, A., Dessì, N., Liberati, D., Pes, B.: Learning Bayesian Classifiers from Gene-Expression MicroArray Data. In: Bloch, I., Petrosino, A., Tettamanzi, A.G.B. (eds.) WILF 2005. LNCS (LNAI), vol. 3849, pp. 297–304. Springer, Heidelberg (2006)
- [12] Bosin, A., Dessì, N., Pes, B.: High-Dimensional Micro-array Data Classification using Minimum Description Length and Domain Expert Knowledge. In: Ali, M., Dapoigny, R. (eds.) IEA/AIE 2006. LNCS (LNAI), vol. 4031, pp. 790–799. Springer, Heidelberg (2006)
- [13] Bosin, A., Dessì, N., Pes, B.: Capturing Heuristics and Intelligent Methods for Improving Micro-Array Data Classification. In: Yin, H., Tino, P., Corchado, E., Byrne, W., Yao, X. (eds.) IDEAL 2007. LNCS, vol. 4881, pp. 790–799. Springer, Heidelberg (2007)
- [14] Armstrong, E., et al.: The J2EE 1.4 Tutorial (2004)
- [15] Oracle database 10g enterprise edition, <http://www.oracle.com>
- [16] <http://taverna.sourceforge.net>
- [17] Oinn, T.: Xscufl Language Reference, European Bioinformatics Institute (2004), <http://www.ebi.ac.uk/~tmo/mygrid/XScuflSpecification.html>
- [18] Golub, T.R., et al.: Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring. Science 286, 531–537 (1999)
- [19] Yeoh, E.J., et al.: Classification, subtype discovery, and prediction outcome in pediatric acute lymphoblastic leukemia by gene expression profiling. Cancer Cell 1 (2002)
- [20] <http://www.dsf.unica.it/~andrea/webservices.html>

Personal Knowledge Network Reconfiguration Based on Brain Like Function Using Self Type Matching Strategy

JeongYon Shim

Div. of GS, Computer Science, Kangnam University
San 6-2, Kugal Dong, KiHeung Gu, YongIn Si, KyeongKi Do, Korea
Tel.: +82 31 2803 736
mariashim@kangnam.ac.kr

Abstract. In the natural brain, memory retaining, reconfiguration and retrieval are very important functions for maintaining the fresh memory. Especially reconfiguration takes an important role for arranging more relevant information to the personal aspect and purpose closely. The closely rearranged information is easy to be activated for processing. In this perspective, the strategy for reconfiguration of Personal Knowledge Network was proposed. Personal Knowledge network is reconstructed by Type matching selection and knowledge reconfiguration algorithm. The proposed system was applied to the Virtual Memory and tested with the sample data.

1 Introduction

Recently the new theory was reported that rearranging process occurs in the real brain when the new learned data comes in and updating process is needed. That is, the connected structure of neural network continues to be updated dynamically. The neural network is composed of neurons and their connections storing the information signals. The structure of personal neural network has a different forms in accordance with personal experience, learning and properties. Though the same input data comes in, it is differently interpreted because the connection structure and properties have a personal difference. In this perspective, it means that a person has an unique personal knowledge network based on the personal characteristics and interests.

As a philosophical concept, there is one concept called as 'Qualia'[1]. It is broadly defined as "The 'What it is like' character of mental states. The way it feels to have mental states such as pain, seeing red, smelling rose, etc.". According to narrower definitions, qualia are ineffable, intrinsic, private and directly or immediately apprehensible in consciousness. This concept can be adopted to define the personal properties. For recent years many studies about 'Qualia' are progressing.

In this paper, Personal Knowledge Network Reconfiguration strategy was designed by Self Type Matching Rule.

First, adopting the partial concept of qualia five Types are defined as Self Type for representing the properties. Also Self Energy defining the inside energy is assigned to the individual knowledge node. Self Type and Self Energy are important factors for deciding the properties of personal knowledge nodes. Using these properties the forms of Personal knowledge node and network are designed. Second, the methods of Knowledge Reconfiguration and retrieval are described. Third, this system was applied to the virtual memory and tested with sample data.

2 A Design of Knowledge Network

2.1 Self Type Matching

Type is defined as a factor representing the property of a thing and is classified to five types, M, F, E, K and S. These five types can be flexibly designed for the application area. We also defined Type matching rule. Type matching rule is used for selecting the knowledge from master Knowledge Network. There are two types of matching relations, Attracting Relation and Rejecting Relation.

Attraction Relation	Attracting degree d_i
$M \oplus \gg F$	$d_1=0.5$
$F \oplus \gg E$	$d_2=0.5$
$E \oplus \gg K$	$d_3=0.5$
$K \oplus \gg S$	$d_4=0.5$
$S \oplus \gg M$	$d_5=0.5$

Rejecting Relation	Rejecting degree d_i
$M \ominus \gg E$	$d_1=-0.5$
$E \ominus \gg S$	$d_2=-0.5$
$S \ominus \gg F$	$d_3=-0.5$
$F \ominus \gg K$	$d_4=-0.5$
$K \ominus \gg M$	$d_5=-0.5$

The matching rule 'M $\oplus \gg(0.5)$ F' means that M type helps F type with attracting degree 0.5. The value d_s of 'M $\oplus \gg(d_s)$ S' is derived from 'M $\oplus \gg(0.5)$ F $\oplus \gg(0.5)$ E $\oplus \gg(0.5)$ K $\oplus \gg(0.5)$ S'. The attracting degree of multiple relation is calculated by the following equation(1).

$$d_s = \begin{cases} \prod_{i=1}^n (-1)^{n+1} d_i & \text{if } Type_i \neq Type_j \\ 1 & \text{otherwise} \end{cases} \tag{1}$$

The value of d_s in 'M $\oplus \gg(d_s)$ S' is 0.0625. If the value of d_s is positive, it is attracting relation. Otherwise, the minus value means rejecting relation.

2.2 The Representation of Knowledge Node and Knowledge Network

• Knowledge node

Knowledge node is an basic atom composing the Knowledge Network. It contains 'Name', 'Type', 'Energy' attributes which can identify itself. Knowledge node is represented as a form of 'struct'.

```
struct  $k - node_i$  (Name, Type, Energy)
```

The term of Energy describes Self Energy value of [-1.0,1.0] inside the individual knowledge node. The minus value means a negative state and the plus value means a positive state. If the value of Self Energy is zero, it is on the neutral point.

• Knowledge Network

Knowledge Network is connected by associative relations between Knowledge nodes and contain the information. It is represented as

```
 $\langle K - node_i, R_{ij}, K - node_j \rangle$ 
```

where $K - node_i$ is the name of knowledge node and R_{ij} is connection strength between two knowledge nodes. R_{ij} is calculated by equation (2).

$$R_{ij} = P(K - node_i | K - node_j) \quad (2)$$

This form is implemented by array and used during the process of Knowledge Network Reconfiguration and Thinking Chain Retrieval.

3 Personal Knowledge Network Reconfiguration

3.1 Knowledge Network Representation

As shown in Figure 1, Personal Knowledge Reconfiguration is the process of extracting the Type matching knowledge nodes from Master Knowledge Network and reconstructing Personal Knowledge Network. Personal Knowledge Reconfiguration has two stages for making the personal aspect reflected structure. First stage is Type Matching Selection. In this stage, with the keyword the system selects Type matched nodes by Type matching rule and calculates matching degree. During this Selection mechanism, the only attracting relations are considered and the rejecting cases are discarded. Second stage is Network Reconfiguration. The selected knowledge nodes start to be connected one by one and their new connection strengths are calculated by equation (3).

$$R_{ij}^{new} = (R_{ij} + d_S) / 2.0 \quad (3)$$

The complete personal knowledge network is used for knowledge retrieval process, inference and decision making.

Knowledge Reconfiguration Mechanism is as following algorithm 1.

Algorithm 1 : Knowledge Reconfiguration

*** Knowledge type matching selection**

STEP1 : Input Keyword.

STEP2 : Search ID of knowledge node matched with the Keyword.

STEP3 : **If** found != true

stop.

Else Select the matched knowledge node and associative relation with Type matching rule.

Calculate the matching degree.

Construct the extracted knowledge Network.

*** Knowledge network concatenation in the personal site.**

STEP4 : Search the matched knowledge node with the Keyword in the previous knowledge network in the personal site.

STEP5 : **If** found != true

Attach the extracted knowledge network to the starting node.

Else Attach the extracted knowledge network to the searched node.

Calculate the new connection strength.

STEP6 : Output the concatenated knowledge network.

STEP7 : Stop.

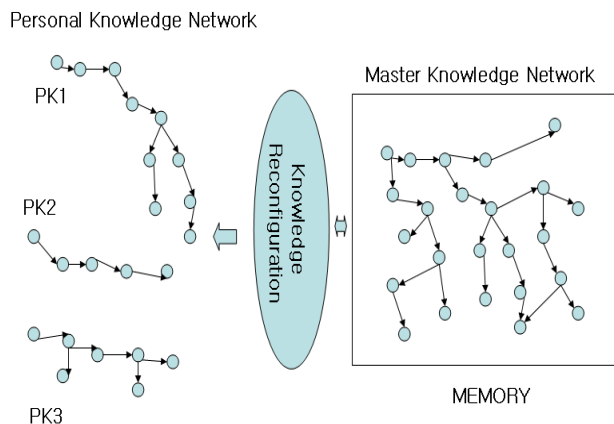


Fig. 1. Personal Knowledge Network Reconfiguration

4 Experiment

This reconfiguration mechanism was applied to the virtual memory and tested with Master knowledge Network frame with virtual knowledge nodes shown in

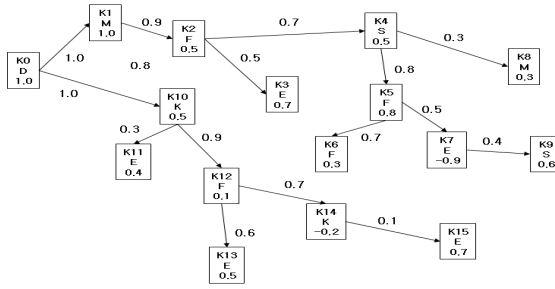


Fig. 2. Master Knowledge Network

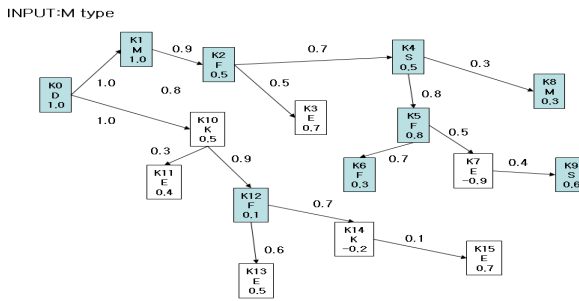


Fig. 3. Personal Knowledge Network : Type M

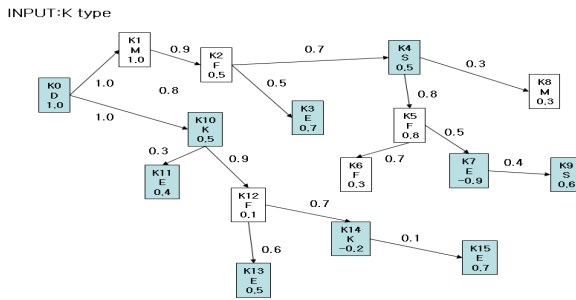


Fig. 4. Personal Knowledge Network : Type K

figure 2. Figure 3 describes activated state by Type Matching Selection about the input Type M. The shaded blocks represent the activated status of knowledge node. In this case only knowledge nodes which has positive value of Type Matching degree are selected for reconfiguration. Figure 4 also shows the activated result of Input Type K. Using the selected knowledge nodes this system produce the result of network reconfiguration mechanism as depicted in Figure 5. As a result it was shown that the knowledge reconfiguration mechanism was

Personal Knowledge Network Reconfiguration
Input Type? M
...Reconfiguration Result
 ...

Knode	Type	Energy	M-T-degree	Rij	Knode
K0	D	1.00000000	1.00000000	1.00000000	K1
K0	D	1.00000000	1.00000000	0.90000000	K12
K1	M	1.00000000	1.00000000	0.90000000	K2
K2	F	0.50000000	0.50000000	0.70000000	K4
K4	S	0.50000000	0.06250000	0.80000000	K5
K4	S	0.50000000	0.06250000	0.30000000	K8
K5	F	0.80000000	0.50000000	0.70000000	K6
K5	F	0.80000000	0.50000000	0.20000000	K9
K6	F	0.30000000	0.50000000	0.00000000	Null
K8	M	0.30000000	1.00000000	0.00000000	Null
K9	S	0.60000000	0.06250000	0.00000000	Null
K12	F	0.10000000	0.50000000	0.00000000	Null

Fig. 5. The result of Reconfiguration :Type M

successfully processed. This reconfiguration mechanism can be used for implementing the concept of personalization in the intelligent system.

5 Conclusions

In this study, adopting the partial concept of qualia Self Type and Energy are defined and Personal Knowledge Network Reconfiguration strategy by Type matching Selection was proposed. It was applied to the virtual memory and tested with virtual knowledge nodes in Virtual Memory. As a result of experiment, the reconfiguration mechanism was successfully processed. This mechanism can be applied to many areas related to intelligent system design and developed to the more efficient mechanism.

References

1. Wikipedia dictionary, <http://en.wikipedia.org/wiki/Qualia>
2. Shim, J.Y.: Knowledge Network Management System with medicine Self Repairing strategy. In: Kang, L., Liu, Y., Zeng, S. (eds.) ICES 2007. LNCS, vol. 4684, pp. 119–128. Springer, Heidelberg (2007)
3. Shim, J.Y.: The design of Self Internal Entropy Balancing System with Incarnation process. In: Li, K., Fei, M., Irwin, G.W., Ma, S. (eds.) LSMS 2007. LNCS, vol. 4688, pp. 55–62. Springer, Heidelberg (2007)
4. Shim, J.Y.: Intelligent capsule Design for the multi functional aspects. In: Li, K., Fei, M., Irwin, G.W., Ma, S. (eds.) LSMS 2007. LNCS, vol. 4688, pp. 719–725. Springer, Heidelberg (2007)
5. Bruce Goldstein, E.: Sensation and Perception, 5th edn. Brooks/Cole Publishing Company (1999)

6. Arbib, M.A., Grethe, J.S.: Computing the brain: A guide to Neuroinformatics. Academic Press, London (2001)
7. Zhong, N., Liu, J.: Intelligent Technologies for Information Analysis. Springer, Heidelberg (2004)
8. Giudici, P.: Applied Data Mining: Statistical Method for Bussiness and Industry. Wiley, Chichester (2003)
9. Schacter, D.L., Scarry, E.: Memory, Brain, and Belief. Harvard University Press, Cambridge (2000)
10. De Raedt, L., Siebes, A.: PKDD 2001. LNCS (LNAI), vol. 2168. Springer, Heidelberg (2001)
11. Cloete, I., Zurada, K.M.: Knowledge Based Neuro Computing. MIT Press, Cambridge (2000)

A Theoretical Derivation of the Kernel Extreme Energy Ratio Method for EEG Feature Extraction

Shiliang Sun

Department of Computer Science and Technology,
East China Normal University, Shanghai 200241, China
s1sun@cs.ecnu.edu.cn

Abstract. In the application of brain-computer interfaces (BCIs), energy features are both physiologically well-founded and empirically effective to describe electroencephalogram (EEG) signals for classifying brain activities. Recently, a linear method named extreme energy ratio (EER) for energy feature extraction of EEG signals in terms of spatial filtering was proposed. This paper gives a nonlinear extension of the linear EER method. Specifically, we use the kernel trick to derive a kernelized version of the original EER feature extractor. The solutions for optimizing the criterion in kernel EER are provided for future use.

Keywords: brain-computer interface (BCI), extreme energy ratio (EER), EEG signal classification, feature extraction, kernel machine.

1 Introduction

A brain-computer interface (BCI) is a direct connection between the brain and external devices, namely independent of peripheral nerves and muscles [12]. BCIs have some latent applications, for example, serving as a communication and control channel for motor-disabled people, alarming paroxysmal diseases for neuropaths, manipulating equipments in inconvenient environments. During recent years, research on BCIs has attracted much interest among different disciplines such as neurophysiology, biomedical engineering and computer science. This paper investigates feature extraction of electroencephalogram (EEG) signals in EEG-based BCIs from the aspect of computer science.

An EEG-based BCI refers to a BCI adopting EEG signals as the information carrier. EEG signals are brain activities recorded by electroencephalography using electrodes mounted on the scalp, which is a convenient and inexpensive means to monitor brain activities. A general EEG-based BCI, as given in Fig. 1, consists of four basic components, which are EEG signal acquisition, feature extraction, pattern classification and device control. Each component is indispensable in order for a BCI to operate successfully. We will study EEG signal feature extraction using machine learning methodologies in this paper. Specifically, energy feature extraction in terms of spatial filtering will be focused on.

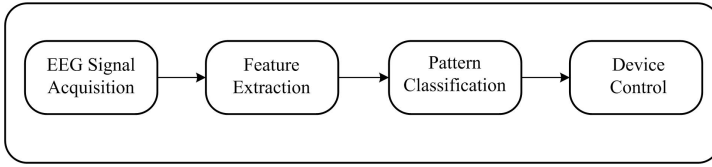


Fig. 1. A block diagram of a general EEG-based BCI

Spatial filtering methods transform EEG signals by combining recordings from different and often adjacent electrodes. Typical spatial filtering methods include Laplacian derivation, common average reference, common spatial patterns (CSP) and extreme energy ratio (EER) [2,3]. The Laplacian derivation derives the second spatial derivative and can be quantified by manipulating the voltages of the current site and its surrounding sites. As the name signifies, the method of common average reference converts voltages employing the mean voltage of all the recording electrodes as a reference. The spatial filtering method of CSP leads to signals which discriminate optimally between two classes of brain activities [4]. There are mainly two operations embodied in the CSP method, namely whitening transform and projection transform. As reported in the literature, using the same classifiers the classification performance obtained by the CSP method is usually as good as, or higher than, those obtained previously by other filtering methods [4].

The recently proposed EER method is theoretically equivalent to CSP, but has an intuitive interpretation and a lower computation burden [3]. Computationally, it only needs a generalized eigenvalue decomposition between two covariance matrices respectively belonging to two different categories. However, EER is still a linear method which may prevent itself from extracting nonlinear energy features. This is the motivation of this paper. Here we provide a nonlinear extension of the EER method called kernel EER by means of the kernel trick [5].

In this paper, we only consider feature extraction for binary classification, since the proposed method can be simply extended from binary to multiclass case by standard one-versus-one or one-versus-rest division. The rest of this paper is organized as follows. Section 2 summarizes the EER method and also gives some necessary information on problem background and variable conventions. Section 3 presents the kernel EER method and its optimization process. Finally, Section 4 gives conclusions and future work directions on possible experiments using the kernel EER method.

2 A Brief Review of the EER Method

Scalp EEG waves reflect inside brain activities which come from some inherent signal sources of neuron clusters lying beneath the surface of the brain cortex [6,7]. Here the process of spatial filtering is assumed to be recovering these latent sources. Energy features of one or multiple sources can be extracted from

each class of brain activities, which are calculated as the variances of these sources.

2.1 One Source from Each Class

The EER method is based on covariances of EEG signals from two different classes denoted as classes A and B (e.g., two different mental imagery tasks). Denote an EEG sample as an $N \times T$ matrix X , where N is the number of electrodes and T is the number of snapshots in the sample. For EEG-signal analysis, the mean value of X can be regarded to be zero as bandpass filters are usually adopted. In addition, to eliminate the energy differences caused due to different recording instants, a normalization is often performed [3]. The estimation for the covariance of one EEG sample can be represented as $C = \frac{1}{T}XX^T$. The covariances C_A and C_B respectively for classes A and B can be obtained as the average of all related covariances calculated from single samples.

Consider only one source from each class is to be recovered with spatial filter ϕ . Then for the EEG sample X , the spatially filtered signal will be $\phi^T X$. The signal energy after filtering can be represented by the sample variance as $\frac{1}{T}\phi^T X(\phi^T X)^T = \phi^T C \phi$. The discriminative EER criterion is defined as

$$R(\phi) \triangleq \frac{\phi^T C_A \phi}{\phi^T C_B \phi}, \quad (1)$$

which indicates the energy ratio after spatial filtering for two classes A and B .

For succeeding classification we can optimize (1) to find the filter ϕ^* which maximizes or minimizes the ratio [3]. Therefore, there are in fact two optimal spatial filters ϕ_{max}^* and ϕ_{min}^* to be sought which satisfy $\phi_{max}^* = \arg \max_{\phi} R(\phi)$ and $\phi_{min}^* = \arg \min_{\phi} R(\phi)$. It turns out that the optimal spatial filters ϕ_{max}^* and ϕ_{min}^* are two eigenvectors respectively corresponding to the maximal and minimal generalized eigenvalues of the matrix pair (C_A, C_B) . Vectors ϕ_{max}^* and ϕ_{min}^* are then normalized to have unit length. For a new EEG sample, its energy feature will be a vector consisting of two entries which are respectively the energy values of the sample spatially filtered by ϕ_{max}^* and ϕ_{min}^* .

2.2 Multiple Sources from Each Class

Recovering only one source from each class is often insufficient to describe the inherent brain activities. Thus, the demand of extracting multiple sources from each class is put forward. For multidimensional signals, since eigenvalues of a covariance matrix can be explained as the variances on principal directions of the corresponding data distribution, the determinant of a covariance matrix represents the product of the signal energies from all of the principal directions. Therefore, determinant can be used to extend the former EER to energy-feature extraction of multiple sources.

Suppose there are m sources to be extracted from each class of brain activities, then the EER method will search for $2m$ sources totally, half of which maximize

the object criterion and the other half minimize it. Let the m spatial filters for extracting m sources constitute a spatial filter bank $\Phi \triangleq [\phi_1, \phi_2, \dots, \phi_m]$. Now the discriminative EER criterion can be written as

$$R(\Phi) \triangleq \frac{|\Phi^\top C_A \Phi|}{|\Phi^\top C_B \Phi|}, \quad (2)$$

where like $R(\phi)$ in [\[10\]](#), $R(\Phi)$ indicates the energy ratio of EEG signals after spatial filtering for two classes A and B . The optimization task is to find the filter bank Φ^* which maximizes or minimizes the ratio. The two optimal spatial filter banks Φ_{max}^* and Φ_{min}^* which give two extreme criterion values should satisfy $\Phi_{max}^* = \arg \max_{\Phi} R(\Phi)$ and $\Phi_{min}^* = \arg \min_{\Phi} R(\Phi)$.

The solution is that Φ_{max}^* consists of m eigenvectors corresponding to the m maximal generalized eigenvalues of the matrix pair (C_A, C_B) , while Φ_{min}^* consists of m eigenvectors whose corresponding generalized eigenvalues are minimal [\[3\]](#). When used for filtering, each column in Φ_{max}^* and Φ_{min}^* is normalized to have unit length. For a new EEG sample, its energy feature can be taken as a vector consisting of $2m$ entries, which are the energy values of the sample respectively filtered by the $2m$ spatial filters coming from two filter banks Φ_{max}^* and Φ_{min}^* .

3 Kernel EER: A Kernelized Extension of EER

The EER method is capable of extracting energy features of EEG signals through the learned linear spatial filters. However, due to this limitation of linearity in the desirable transform, EER can not guide the learning of nonlinear spatial filters for energy-feature extraction. Since the nonlinear combination of signals from different electrodes may provide features which are more discriminative, or there may be a latent nonlinear structure in the signal distribution, it makes sense for us to devise nonlinear spatial filters for extracting energy features of EEG signals.

We propose to adopt the kernel trick [\[8\]](#), which is widely applied such as in kernel principal component analysis [\[9\]](#), kernel Fisher discriminant analysis [\[10\]](#), and support vector machines [\[11\]](#), to define a nonlinear generalization of the original EER method. In other words, the optimal spatial filters will be looked for in a kernel space \mathcal{F} , which is related to the original space \mathbb{R}^N by a possibly nonlinear map Ψ

$$\mathbb{R}^N \rightarrow \mathcal{F}, \quad x \mapsto \Psi(x).$$

One important feature of the kernel trick is that it adopts kernel functions (Mercer kernels) to facilitate the calculation of dot products in the kernel space with signal representations in the original space [\[12\]](#). For instance, for a kernel function $k(\cdot, \cdot)$ and two vectors x and y from the space \mathbb{R}^N , we have $(\Psi(x) \cdot \Psi(y)) = k(x, y)$. Typical kernel functions include radial basis functions, polynomial functions, and sigmoid kernels [\[11\]](#).

We give some new notations. Define $X_{A_p} = \{x_1^{A_p}, \dots, x_T^{A_p}\}$ ($p = 1, \dots, t_A$) and $X_{B_q} = \{x_1^{B_q}, \dots, x_T^{B_q}\}$ ($q = 1, \dots, t_B$) to be samples from two different

classes A and B with t_A and t_B being their corresponding sample numbers. One sample is composed of T N -dimensional snapshots (one snapshot is the EEG recording at an instant of time). The total snapshot set is defined as $\mathcal{X} = \{X_{A_p}\} \cup \{X_{B_q}\} = \{x_1, \dots, x_l\}$ with $l = T(t_A + t_B)$. The covariances of signals from A_p and B_q in the kernel space are respectively computed as

$$\begin{aligned} C_{A_p}^\Psi &= \frac{1}{T} \sum_{i=1}^T (\Psi(x_i^{A_p}) - m_{A_p}^\Psi)(\Psi(x_i^{A_p}) - m_{A_p}^\Psi)^\top, \\ C_{B_q}^\Psi &= \frac{1}{T} \sum_{i=1}^T (\Psi(x_i^{B_q}) - m_{B_q}^\Psi)(\Psi(x_i^{B_q}) - m_{B_q}^\Psi)^\top, \end{aligned} \tag{3}$$

where m_j^Ψ ($j = \{A_p, B_q\}$) is the one-sample mean with $m_j^\Psi = \frac{1}{T} \sum_{i=1}^T \Psi(x_i^j)$. Because the possibly high dimensionality of the kernel space, the covariances are usually not computed explicitly but represented by the images of individual snapshots for the convenience of further processing.

The signal covariances from classes A and B in the kernel space can be represented as

$$C_A^\Psi = \frac{1}{t_A} \sum_{p=1}^{t_A} C_{A_p}^\Psi, \quad C_B^\Psi = \frac{1}{t_B} \sum_{q=1}^{t_B} C_{B_q}^\Psi. \tag{4}$$

3.1 One Source from Each Class

As a nonlinear generalization of (11), we define the discriminative criterion of kernel EER for learning optimal nonlinear spatial filters as

$$R^\Psi(\omega) \triangleq \frac{\omega^\top C_A^\Psi \omega}{\omega^\top C_B^\Psi \omega}, \tag{5}$$

where $\omega \in \mathcal{F}$, and $R^\Psi(\omega)$ indicates the energy ratio after spatial filtering for two classes A and B in the high dimensional space \mathcal{F} . It is acknowledged that any significant ω must lie in the span of the images of all given snapshots [5]. Hence, ω can be expressed in the following form

$$\omega = \sum_{i=1}^l \alpha_i \Psi(x_i), \tag{6}$$

where α_i ($i = 1, \dots, l$) are scalars.

Based on (4) and (6), the numerator in (5) can be rewritten as

$$\omega^\top C_A^\Psi \omega = \frac{1}{t_A} \sum_{p=1}^{t_A} \omega^\top C_{A_p}^\Psi \omega. \tag{7}$$

And $\omega^\top C_{A_p}^\Psi \omega$ can be reformulated as

$$\omega^\top C_{A_p}^\Psi \omega = \frac{1}{T} \omega^\top \sum_{i=1}^T (\Psi(x_i^{A_p}) - m_{A_p}^\Psi)(\Psi(x_i^{A_p}) - m_{A_p}^\Psi)^\top \omega$$

$$\begin{aligned}
 &= \frac{1}{T} \omega^\top \sum_{i=1}^T (\Psi(x_i^{A_p}) \Psi^\top(x_i^{A_p}) - m_{A_p}^\Psi (m_{A_p}^\Psi)^\top) \omega \\
 &= \frac{1}{T} \alpha^\top N_{A_p} \alpha,
 \end{aligned} \tag{8}$$

where $\alpha^\top = [\alpha_1, \dots, \alpha_l]$, $N_{A_p} = K_{A_p} (I - 1_T) K_{A_p}^\top$, $(K_{A_p})_{l \times T}$ is the kernel matrix for the class A with $(K_{A_p})_{ij} = k(x_i, x_j^{A_p}) = (\Psi(x_i) \cdot \Psi(x_j^{A_p}))$, I is the identity matrix and 1_T is the matrix with each element being $1/T$ [10]. Define $N_A = \frac{1}{t_A} \sum_{p=1}^{t_A} N_{A_p}$, we have

$$\omega^\top C_A^\Psi \omega = \frac{1}{T} \alpha^\top N_A \alpha. \tag{9}$$

Analogously, the denominator in (5) can be obtained as

$$\omega^\top C_B^\Psi \omega = \frac{1}{T} \alpha^\top N_B \alpha, \tag{10}$$

where the specific formulation of N_B can be naturally transferred from that of N_A .

Combining (9) and (10), the kernel EER criterion can be equivalently simplified as

$$R_\Psi(\alpha) \triangleq \frac{\alpha^\top N_A \alpha}{\alpha^\top N_B \alpha}.$$

On the analogy of solving EER, the solutions α_{max} and α_{min} that maximize and minimize $R_\Psi(\alpha)$ can be given as the two eigenvectors respectively corresponding to the maximal and minimal generalized eigenvalues for the matrix pair (N_A, N_B) . Accordingly, the optimal kernel spatial filters ω_{max} and ω_{min} are obtained in terms of (6).

It should be noted that in order to make ω in (6) have unit length, constraints about α should be added. Formally, substituting (6) into $\omega^\top \omega = 1$ gives

$$\left[\sum_{i=1}^l \alpha_i \Psi^\top(x_i) \right] \left[\sum_{j=1}^l \alpha_j \Psi(x_j) \right] = 1, \tag{11}$$

that is

$$\alpha^\top K_l \alpha = 1, \tag{12}$$

where K_l is an $l \times l$ matrix with $(K_l)_{ij} = k(x_i, x_j)$.

For a new EEG sample Y consisting of T snapshots, that is

$$Y = [y_1, \dots, y_T],$$

the filtered sample by the nonlinear spatial filter ω (for example, ω_{max} or ω_{min}) will be

$$\omega^\top \Psi(Y) = \sum_{i=1}^l \alpha_i \Psi^\top(x_i) [\Psi(y_1), \dots, \Psi(y_T)]$$

$$\begin{aligned}
 &= \left[\sum_{i=1}^l \alpha_i (\Psi(x_i) \cdot \Psi(y_1)), \dots, \sum_{i=1}^l \alpha_i (\Psi(x_i) \cdot \Psi(y_T)) \right] \\
 &= \left[\sum_{i=1}^l \alpha_i k(x_i, y_1), \dots, \sum_{i=1}^l \alpha_i k(x_i, y_T) \right].
 \end{aligned}$$

The variance of $\omega^\top \Psi(Y)$ is the extracted energy feature. Therefore, the energy-feature vector of Y after nonlinear spatial filtering includes two entries, which are respectively the variances of $\omega_{max}^\top \Psi(Y)$ and $\omega_{min}^\top \Psi(Y)$.

3.2 Multiple Sources from Each Class

To generalize (2) to nonlinear spatial filtering, that is, to extract m sources from each class of brain activities in the kernel space, a kernel spatial filter bank $\Omega \triangleq [\omega_1, \dots, \omega_m]$ should be learned for each class where $\omega_j \in \mathcal{F}$ ($j = 1, \dots, m$). Consulting the former explanation of determinant for describing energy, the discriminative kernel EER criterion can be defined as

$$R^\Psi(\Omega) \triangleq \frac{|\Omega^\top C_A^\Psi \Omega|}{|\Omega^\top C_B^\Psi \Omega|}. \tag{13}$$

Let $\alpha_j^\top = [\alpha_1^j, \dots, \alpha_l^j]$ ($j = 1, \dots, m$) be possible coefficient vectors. Similar to (6) we can find the expansion for ω_j using α_j

$$\omega_j = \sum_{i=1}^l \alpha_i^j \Phi(x_i). \tag{14}$$

Suppose $\mathcal{Z} = [\alpha_1, \dots, \alpha_m]$ is a matrix composed of all the coefficient vectors. Repeating the operations as in (7) ~ (10), we can get a formulation equivalent to the criterion of (13)

$$R_\Psi(\mathcal{Z}) \triangleq \frac{|\mathcal{Z}^\top N_A \mathcal{Z}|}{|\mathcal{Z}^\top N_B \mathcal{Z}|}.$$

By analogy with solutions for EER, the matrices \mathcal{Z}_{max} and \mathcal{Z}_{min} that maximize and minimize $R_\Psi(\mathcal{Z})$ are trivial to derive. Namely, \mathcal{Z}_{max} consists of m eigenvectors of the matrix pair (N_A, N_B) which correspond to the m maximal generalized eigenvalues, while \mathcal{Z}_{min} consists of m eigenvectors whose corresponding generalized eigenvalues are minimal. Accordingly, the optimal kernel spatial filter banks Ω_{max} and Ω_{min} which optimize $R^\Psi(\Omega)$ can be derived in terms of (14). Of course, each column in \mathcal{Z}_{max} and \mathcal{Z}_{min} should be normalized as in (11) and (12) in order to make ω_j ($j = 1, \dots, m$) be unit vectors. We can see that when $m = 1$, (13) degenerates to (5) with the same solutions and thus (5) is a special case of (13).

The m -dimensional signal after spatial filtering by Ω (for example, Ω_{max} or Ω_{min}) for an EEG snapshot y_j ($j = 1, \dots, T$) from the sample Y is

$$\Omega^\top \Psi(y_j) = [\omega_1, \dots, \omega_m]^\top \Psi(y_j)$$

$$\begin{aligned}
&= [\omega_1^\top \Psi(y_j), \dots, \omega_m^\top \Psi(y_j)]^\top \\
&= \left[\sum_{i=1}^l \alpha_i^1 k(x_i, y_j), \dots, \sum_{i=1}^l \alpha_i^m k(x_i, y_j) \right]^\top. \tag{15}
\end{aligned}$$

Thereby, the filtered signal for Y can be obtained by applying (15) for every snapshot

$$\Omega^\top \Psi(Y) = [\Omega^\top \Psi(y_1), \dots, \Omega^\top \Psi(y_T)].$$

For feature extraction with intent to classification, given the input Y we actually obtain a $2m$ -dimensional filtered signal concatenated by $\Omega_{max}^\top \Psi(Y)$ and $\Omega_{min}^\top \Psi(Y)$. The variance on each dimension serves as a feature, and the complete feature vector includes $2m$ such entries.

4 Conclusion

In this paper, a nonlinear energy feature extractor named kernel EER for learning optimal spatial filters for classification is proposed. It is a kernelized version of the linear EER method. We provide details for solution solving in kernel EER.

The contribution of this paper is on theoretical derivation of the kernel EER method. As an important future work, we will conduct experiments with various kernel functions to evaluate the performance of the kernel EER method.

Acknowledgments. This work is supported in part by the National Natural Science Foundation of China under Project 60703005, and in part by Shanghai Educational Development Foundation under Project 2007CG30.

References

1. Nicolelis, M.A.L.: Actions from Thoughts. *Nature* 409, 403–407 (2001)
2. Wolpaw, J.R., Birbaumer, N., McFarland, D.J., Pfurtscheller, G., Vaughan, T.M.: Brain-Computer Interfaces for Communication and Control. *Clin. Neurophysiol.* 113, 767–791 (2002)
3. Sun, S.: The Extreme Energy Ratio Criterion for EEG Feature Extraction. In: Proc. 18th Int. Conf. Artificial Neural Networks, Prague, Czech Republic (2008)
4. Müller-Gerking, J., Pfurtscheller, G., Flyvbjerg, H.: Designing Optimal Spatial Filters for Single-Trial EEG Classification in a Movement Task. *Clin. Neurophysiol.* 110, 787–798 (1999)
5. Müller, K.R., Mika, S., Rättsch, G., Tsuda, K., Schölkopf, B.: An Introduction to Kernel-Based Learning Algorithms. *IEEE Trans. Neural Netw.* 12, 181–201 (2001)
6. Curran, E.A., Stokes, M.J.: Learning to Control Brain Activity: A Review of the Production and Control of EEG Components for Driving Brain-Computer Interface (BCI) Systems. *Brain Cogn.* 51, 326–336 (2003)
7. Kamousi, B., Liu, Z., He, B.: Classification of Motor Imagery Tasks for Brain-Computer Interface Applications by Means of Two Equivalent Dipoles Analysis. *IEEE Trans. Neural Syst. Rehabil. Eng.* 13, 166–171 (2005)

8. Aizerman, M., Braverman, E., Rozonoer, L.: Theoretical Foundations of the Potential Function Method in Pattern Recognition Learning. *Automation and Remote Control* 25, 821–837 (1964)
9. Schölkopf, B., Smola, A., Müller, K.R.: Nonlinear Component Analysis as a Kernel Eigenvalue Problem. *Neural Comput.* 10, 1299–1319 (1998)
10. Mika, S., Rätsch, G., Weston, J., Schölkopf, B., Müller, K.R.: Fisher Discriminant Analysis with Kernels. In: *Proc. IEEE Int. Workshop Neural Networks for Signal Processing IX*, Madison, USA, pp. 41–48 (1999)
11. Vapnik, V.: *The Nature of Statistical Learning Theory*, 2nd edn. Springer, New York (2000)
12. Schölkopf, B., Smola, A.: *Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond*. MIT Press, Cambridge (2002)

Control of a Wheelchair by Motor Imagery in Real Time

Kyuwan Choi and Andrzej Cichocki

BSI-TOYOTA Collaboration Center, RIKEN
kwchoi@brain.riken.jp

Abstract. This paper gives an outline of a non-invasive brain machine interface (BMI) implemented for controlling a motorized wheelchair online. Subjects were trained by using an effective feedback training method, and they could then control the wheelchair freely, similar to controlling it with a joystick.

Keywords: BMI, EEG, Wheelchair, Feedback training.

1 Introduction

One of the most processing paradigms in BMI is motor imagery paradigm [2] in which the mu (8-13 Hz) and beta rhythm (14-30 Hz) of the sensorimotor cortex are used. The oscillations of the mu and beta rhythm of the sensorimotor cortex decrease when a movement is being prepared or during a movement—this is called event related desynchronization (ERD). After a movement occurs, the oscillations increase—this is called event related synchronization (ERS). If a person imagines that s/he is moving the left hand, a strong ERD occurs at the right side of the sensorimotor cortex. On the other hand, if a person imagines that s/he is moving the right hand, ERD occurs at the left side. In foot imagery, ERD occurs in the foot area of the sensorimotor cortex. By using this phenomenon, it is possible to control an object by extracting commands from brain signals. In the case of motor imagery paradigm, the subject is not exposed to any stimulation; therefore, there is no risk of fatigue. However, considerable amount of time and effort is required to train the subjects, and generally, the accuracy is lower than other paradigms such as P300 or SSVEP. Thus, the key points of the motor imagery paradigm are (i) how to train the subjects with minimum time and effort, and (ii) how to obtain high performance.

In this study, through effective signal processing methods and a feedback training method, we minimized the training time and effort required by the subjects and maximized the accuracy. The trained subjects could freely control a wheelchair with multi degrees of freedom and requiring fast response time by using the motor imagery paradigm.

2 Materials and Methods

2.1 Construction of Wheelchair System

The wheelchair system used in this study consists of several processing units (see Figure 1(a)) and operates as follows. It contains the signal acquisition block that is used to measure the electroencephalograph (EEG) signals from the brain (the EEG system used in this study was obtained from GTEC technologies located in Austria). It also contains a signal preprocessing block for artifact rejection and noise reduction (we used the blind source separation (BSS) method). For extracting the features of the signals, it contains spatial filters that can efficiently detect the ERD phenomenon. Linear support vector machines (SVMs) are then used to classify the signals after extracting the features. The classified signals are then transformed to the command output that is used to control the wheelchair. Finally, the wheelchair movement provides feedback to the user. In this system, the wheelchair turns to the left, if the user imagines clenching the left hand. Conversely, it turns to the right, if the user imagines squeezing the right hand. If the user imagines walking with both feet, the wheelchair moves forward.

2.2 Signal Acquisition

EEG signals were referentially recorded using the G-TEC system with 5 Ag/AgCl electrodes placed over the primary motor cortex—the region related to hand and foot motor imagery (See Figure 1(b)). The reference electrode was mounted on the right ear and the grounding electrode was mounted on the forehead (Fpz). The sampling frequency rate was 256 Hz. The EEG signals were bandpass-filtered between 8 Hz and 30 Hz, and a 50 Hz notch filter was applied to reject the AC artifacts. All electrode impedances were maintained below 10 k Ω . The measured signals then entered the overlapping sliding windows with a length of 1 s for providing continuous feedback. Since the sliding time of the windows was 125 ms, the wheelchair could receive a new command every 125 ms.

2.3 Signal Preprocessing

A second-order BSS algorithm was applied to enhance the signal and to attenuate the artifacts. For the BSS procedure, we applied a modified and improved real-time AMUSE algorithm [3] since such an algorithm enables a very rapid and reliable estimation of independent components with automatic ranking (sorting) according to their increasing frequency contents and/or decreased linear predictability. The AMUSE algorithm can be considered as 2 consecutive PCAs. One PCA is applied to the input data and the next PCA (SVD) is applied to the time-delayed covariance matrix of the output from the previous stage. In the first step, standard or robust prewhitening (sphering) was applied as a linear transformation as follows;

$$\mathbf{z}(t) = \mathbf{Q}\mathbf{x}(t), \quad (1)$$

where $\mathbf{Q} = \mathbf{R}_x^{-\frac{1}{2}}$ of the standard covariance matrix. Further,

$$\mathbf{R}_x = E\{\mathbf{x}(t)\mathbf{x}^T(t)\} \quad (2)$$

and $\mathbf{x}(t)$ is a vector of the observed data for time instant t . Next, SVD was applied to a time-delayed covariance matrix of the pre-whitened data as follows:

$$\mathbf{R}_z = E\{\mathbf{z}(t)\mathbf{z}^T(t-1)\} = \mathbf{U}\Sigma\mathbf{V}^T \quad (3)$$

where Σ is a diagonal matrix with decreasing singular values and \mathbf{U} and \mathbf{V} are matrices of eigenvectors. Then, a demixing (separating) matrix was estimated as follows:

$$\mathbf{W} = \mathbf{U}^T\mathbf{Q}. \quad (4)$$

The estimated independent components were obtained as follows:

$$\mathbf{Y} = \mathbf{W}\mathbf{X}, \quad (5)$$

where $\mathbf{X} = [\mathbf{x}(1), \mathbf{x}(2), \dots, \mathbf{x}(N)]$.

The AMUSE BSS algorithm enabled us in automatically ranking the EEG components. The undesired components corresponding to the artifacts were removed and the remaining useful (significant) components were projected back to the scalp level by using the pseudo inverse of \mathbf{W} . The enhanced EEG signals ($\hat{\mathbf{X}}$), which are the input of the wheelchair system, were obtained as follows:

$$\hat{\mathbf{X}} = \mathbf{W}^+\mathbf{Y}. \quad (6)$$

2.4 Feature Extraction

We used spatial filters to extract the features that distinguish each data group optimally. The filters were obtained by a common spatial pattern (CSP) method [6] from the multichannel EEG signals. Computationally, this method is parallel by nature and requires only scalar products. Therefore, this method is optimal for real-time applications.

The enhanced EEG data of a single trial is represented as an $N \times T$ matrix $\hat{\mathbf{X}}$, where N is the number of channels (i.e., recording electrodes) and T is the number of samples per channel. The normalized spatial covariance of EEG can be obtained from

$$\mathbf{C} = \frac{\hat{\mathbf{X}}\hat{\mathbf{X}}^T}{\text{trace}(\hat{\mathbf{X}}\hat{\mathbf{X}}^T)} \quad (7)$$

where T denotes the transpose operator and $\text{trace}(\mathbf{X})$ is the sum of the diagonal elements of \mathbf{X} . For the 2 distributions to be separated (e.g., left and right movement imagery), the spatial covariance $\overline{\mathbf{C}}_{d \in [a,b]}$ is calculated by averaging the trials of each group. The composite spatial covariance matrix is as follows:

$$\mathbf{C}_c = \overline{\mathbf{C}}_a + \overline{\mathbf{C}}_b. \quad (8)$$

\mathbf{C}_c can be factored as $\mathbf{C}_c = \mathbf{U}_c \lambda_c \mathbf{U}_c^T$, where \mathbf{U}_c is the matrix of eigenvectors and λ_c is the diagonal matrix of eigenvalues. Note that throughout this section, the eigenvalues have been assumed to be sorted in the descending order.

The whitening transformation

$$\mathbf{P} = \sqrt{\lambda^{-1}} \mathbf{U}_c^T \quad (9)$$

equalizes the variances in the space spanned by \mathbf{U}_c , i.e., all eigenvalues of $\mathbf{P} \mathbf{C}_c \mathbf{P}^T$ are equal to 1. If $\bar{\mathbf{C}}_a$ and $\bar{\mathbf{C}}_b$ are transformed as

$$\mathbf{S}_a = \mathbf{P} \bar{\mathbf{C}}_a \mathbf{P}^T \text{ and } \mathbf{S}_b = \mathbf{P} \bar{\mathbf{C}}_b \mathbf{P}^T \quad (10)$$

then \mathbf{S}_a and \mathbf{S}_b share common eigenvectors, i.e.,

$$\text{if } \mathbf{S}_a = \mathbf{B} \lambda_a \mathbf{B}^T \text{ then } \mathbf{S}_b = \mathbf{B} \lambda_b \mathbf{B}^T \text{ and } \lambda_a + \lambda_b = \mathbf{I} \quad (11)$$

where \mathbf{I} is the identity matrix. Since the sum of the 2 corresponding eigenvalues is always equal to 1, the eigenvector with the largest eigenvalue for $\bar{\mathbf{S}}_a$ has the smallest eigenvalue for $\bar{\mathbf{S}}_b$ and vice versa. This property renders the eigenvector \mathbf{B} useful for the classification of the 2 distributions. The projection of the whitened EEG onto the first and last eigenvectors in \mathbf{B} (i.e., the eigenvectors corresponding to the largest λ_a and λ_b) provides feature vectors that are optimal for distinguishing between the 2 populations of the EEG in the least-squares sense.

With the projection matrix $\mathbf{W} = (\mathbf{B}^T \mathbf{P})^T$, the decomposition (mapping) of a trial $\hat{\mathbf{X}}$ is given as follows:

$$\mathbf{Z} = \mathbf{W} \hat{\mathbf{X}}. \quad (12)$$

The signals $\mathbf{Z}_p (p = 1 \cdots 2m)$ that maximize the difference in variance between the 2 groups are those associated with the largest eigenvalues λ_a and λ_b . These signals are the m first and last rows of \mathbf{Z} by the calculation of \mathbf{W}

$$\mathbf{f}_p = \log\left(\frac{\text{var}(\mathbf{Z}_p)}{\sum_{i=1}^{2m} \text{var}(\mathbf{Z}_i)}\right). \quad (13)$$

After selecting \mathbf{Z}_p , the normalized variance was calculated. Next, we obtained the features \mathbf{f}_p that are used to calculate a linear classifier by taking log to have normal distribution. The CSP method can be used for classifying only 2 groups of data. Therefore, in this study, for classifying 3 groups of data, we used the total CSP features that are the summation of the CSP features of each of the 2 groups. The total CSP features of the 3 groups were used at the input of a linear classifier.

2.5 Classification

We used the linear SVMs for classifying the feature vectors obtained from the EEG data into each class of motor imagery. The basic idea of the SVMs classification is to find such a separating hyperplane that corresponds to the largest

possible margin between the points of different classes. This then led to the following learning algorithm for linear SVMs. To enable the classifier to correctly classify the training data points x_1, \dots, x_n with labels y_1, \dots, y_n drawn from $+1/-1$, the following constraints need to be satisfied:

$$w \cdot x_i + b + \xi_i \geq 1 \quad \text{if } y_i = 1 \quad (14)$$

$$w \cdot x_i + b + \xi_i \leq -1 \quad \text{if } y_i = -1 \quad (15)$$

where ξ_i is the distance of the misclassified points from the hyperplane.

2.6 Stop Command Using EMG Signal

Controlling of a wheelchair only by EEG signals is not always perfect; therefore, we need a more reliable channel for stopping the wheelchair. We employed muscle activity for this purpose. The EMG signal was measured at a cheek muscle, and the wheelchair was made to reliably stop with this signal. The EMG signal at the cheek muscle was sampled at 256 Hz with 12-bit resolution. The signal was digitally rectified, averaged for 5 ms, and filtered through a 2nd-order low pass filter with a cut-off frequency of approximately 3 Hz.

$$f_{EMG}(t) = \sum_{j=1}^n h_j EMG(t - j + 1) \quad (16)$$

$$h(t) = 6.44 \times (\exp^{-10.80t} - \exp^{-16.52t}) \quad (17)$$

The coefficient h_j in (16) can be acquired by digitizing $h(t)$ in (17) discretely. The resulting signal is closely similar to the actual tension; therefore, it is called quasi-tension ([1], [4], [5]). A threshold value is first decided. If the EMG value exceeds the threshold value, the wheelchair will be stopped.

2.7 Experiments

Three healthy men (age: 27-33 years) participated in the experiments. Each subject sat in front of a computer and performed an imaginary movement of the hand and foot. When an arrow appeared on the monitor, the subject performed 1 of 3 imaginary movement based on the direction of the arrow. If the arrow pointed to the right, the subject imagined clenching the right hand. Similarly, if the arrow pointed to the left, the subject imagined squeezing the left hand. Further, when the arrow pointed upward, the subject imagined walking with both feet.

When there was a blank page on the computer monitor for 2 s (see Figure 1(c)), the subject relaxed. Further, when an arrow appeared on the monitor, the subject performed an imaginary movement based on the direction for 3 s. We defined 5 s as one trial. After defining 30 trials as 1 set, we performed 7 sets of experiments for each subject. The EEG signals obtained after performing the

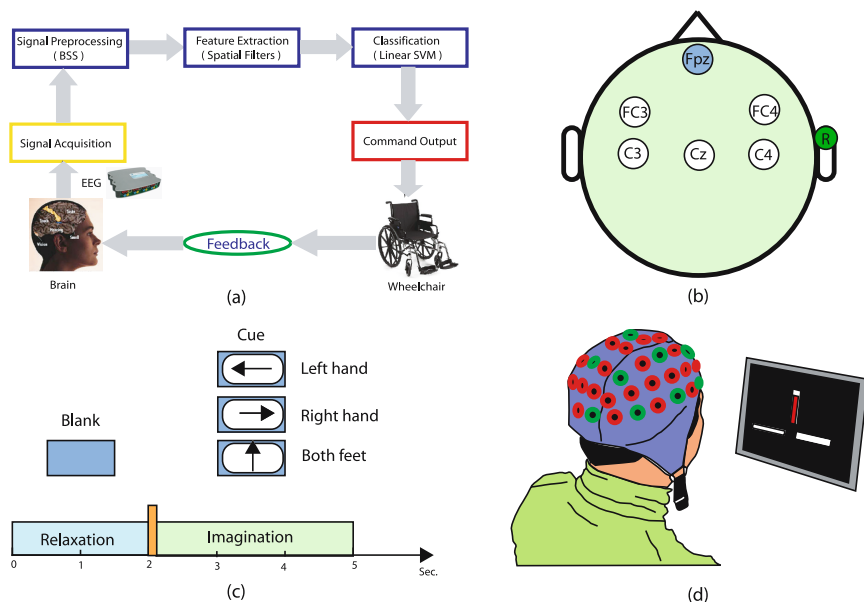


Fig. 1. (a) Conceptual block diagram of the wheelchair controlled with EEG signals. (b) Electrode placement. (c) Imaginary movement task. (d) Visual feedback training.

first set were used to train the parameters of the spatial filter and the linear SVMs. Next, from the second set, the parameters of the spatial filter and the linear SVMs were updated for every set and visual feedback was provided every 125 ms, which represents the probability of selecting each case, to the subject to help imaginary movement, as shown in Figure 1 (d).

3 Results

After completing the experiment, subject 1 who showed a good result was asked to sit on the implemented wheelchair (figure 3 (a)) and drive the wheelchair on an 8-shape course while avoiding 2 obstacles, as shown in Figure 3 (b). He continued driving the course 10 times and could efficiently control the wheelchair without colliding against any obstacles or the wall. On an average, approximately 22.88 s (*standarddeviation* \pm 0.16s) was the time required to drive the course. Further, when driving the course with a joystick, the time required was 16.96 s (*standarddeviation* \pm 0.086s).

Figure 2 (a) represents the error rate for each set. The error rate decreased with every set. Subject 1 adapted to the experiment at an early stage and showed stable results after set 4. Subjects 2 and 3 did not adapt to the experiment at an early stage like subject 1; however, the error rate decreased with every set.

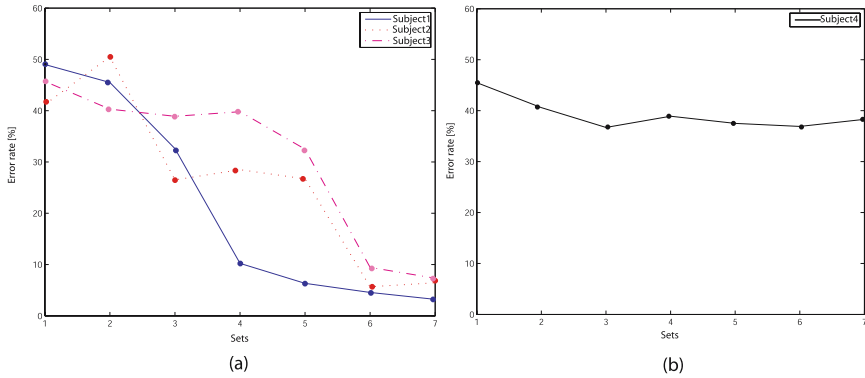


Fig. 2. (a) Error rates of each set of experiment with visual feedback. (b) Error rates of each set without visual feedback.

4 Discussion

Figure 3 (c) represents the most significant spatial patterns (i.e., the first and last columns of \mathbf{W}^{-1} in eq. 12) for the well-trained subject 1. The first row in the figure represents the spatial pattern when the subject imagined clenching the left (L) and right (R) hands. When the subject imagined squeezing the left hand, increased EEG signals were observed at the left side of the sensorimotor cortex. On the other hand, decreased EEG signals were observed at the right side. When the subject imagined clenching the right hand, increased EEG signals were observed over electrodes C4 and FC4 located at the right side of the sensorimotor cortex. The second row in the figure represents the spatial pattern when the subject imagined clenching the right hand and walking with both feet. The EEG signals increased at the right side of the sensorimotor cortex when the subject imagined clenching the right hand. When the subject imagined walking with both feet, the EEG signals over the electrode Cz located at the foot area of the sensorimotor cortex decreased and strong EEG signals occurred at both sides of the sensorimotor cortex. The third row in the figure indicates the spatial pattern when the subject imagined walking with both feet and squeezing the left hand. When the subject imagined walking with both feet, increased EEG signals were observed over electrodes C3 and FC3 located at the left side of the sensorimotor cortex. Further, strong EEG signals were observed over electrodes C4 and FC4 at the right side. When the subject imagined clenching the left hand, the EEG signals were stronger over the broad area of the left side of the brain. The above results are well-matched with the ERD phenomenon.

Figure 2 (b) shows the error rate for subject 4 who carried out the experiment without any visual feedback. Each set was completed and the EEG signals obtained from each set were then used to update the parameters of the spatial filter and the linear SVMs, similar to the experiment using the visual feedback. As observed in Figure 2 (b), the error rate decreased by set 3 but not after that. When these error rates are compared with the error rates in Figure 2 (a)

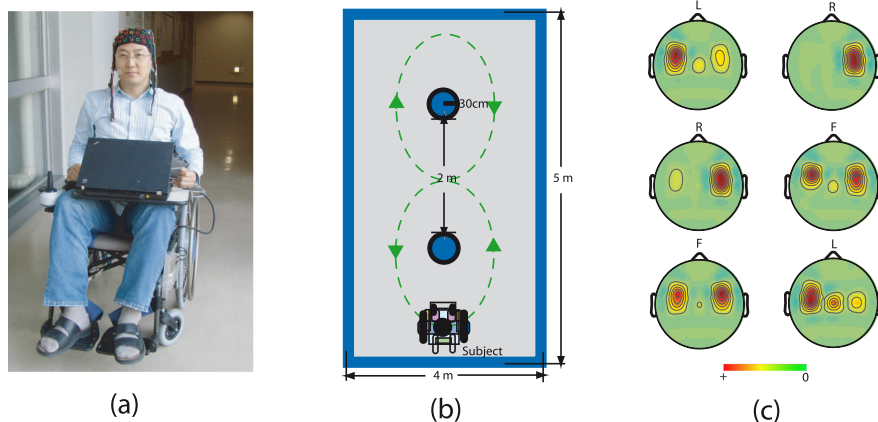


Fig. 3. (a) An actually implemented wheelchair system. (b) Wheelchair obstacle course. (c) The most significant spatial patterns for subject 1.

(with visual feedback), the difference is evident. When visual feedback is provided every 0.125 s, the subjects can adapt to the experiment more easily and within a short period of time. The human brain has the ability to change the neural activities while learning new things. This is known as neuroplasticity. The results of the above experiment indicate that when the visual feedback training method is used, neuroplasticity can occur more easily and within a short period of time.

References

1. Choi, K., Hirose, H., Sakurai, Y., Iijima, T., Koike, Y.: Prediction of arm trajectory from the neural activities of the primary motor cortex using a modular artificial neural network model. In: Ishikawa, M., Doya, K., Miyamoto, H., Yamakawa, T. (eds.) *ICONIP 2007, Part II. LNCS*, vol. 4985, pp. 987–996. Springer, Heidelberg (2008)
2. Wolpaw, J.R., McFarland, D.J.: Control of a two-dimensional movement signal by a noninvasive brain-computer interface in humans. *PNAS* 101, 17849–17854 (2004)
3. Cichocki, A., Amari, S.: *Adaptive blind signal and image processing*, vol. 1. Wiley, Chichester (2002)
4. Koike, Y., Kawato, M.: Estimation of dynamic joint torques and trajectory formation from surface electromyography signals using a neural network model. *Biol. Cybernet* 73, 291–300 (1995)
5. Choi, K., Sato, M., Koike, Y.: Consideration of the Embodiment of a New, Human-centered Interface. *IEICE Trans. Inf. Syst.* E89-D(6) (June 2006)
6. Blankertz, B., Kawanabe, M., Tomioka, R., Hohlefeld, F., Nikulin, V., Müller, K.R.: Invariant common spatial patterns: Alleviating nonstationarities in brain-computer interfacing. In: *Advances in Neural Information Processing Systems*, vol. 20, MIT Press, Cambridge (2008)

Robust Vessel Segmentation Based on Multi-resolution Fuzzy Clustering

Gang Yu¹, Pan Lin², and Shengzhen Cai²

¹ School of Info-Physics and Geometrics Engineering, Central South University, Hunan 410083 China

yugang@mail.csu.edu.cn

² Faculty of Software, Fujian Normal University, Fujian 350007 China

Abstract. A novel multi-resolution approach is presented for vessel segmentation using multi-scale fuzzy clustering and vessel enhancement filtering. According to geometric shape analysis of the vessel structure with different scale, a new fuzzy inter-scale constraint based on antistrophic diffusion linkage model is introduced which builds an efficient linkage relationship between the high resolution feature images and low resolution ones. Meanwhile, this paper develops two new fuzzy distances which describe the fuzzy similarity of line-like structure in adjacent scales effectively. Moreover, a new multiresolution framework combining the inter- and intra-scale constraints is presented. The proposed framework is robust to noisy vessel images and low contrast ones, such as medical images. Segmentation of a number of vessel images shows that the proposed approach is robust and accurate.

1 Introduction

Vessel segmentation is an important area in medical image processing. The vessel segmentation methods were widely investigated in the past. Early conventional approaches for vessel segmentation are matched filter methods [1] and morphological methods [2]. In these approaches, detection accuracy and validity of post processing are undesirable. Recently, active contour (snake) models [5-8] have become effective tools for extraction of the region of interests(ROI), which were widely used for vessel segmentation, but the methods are almost computation expensive.

Recently, Multiscale or multiresolution approaches for medical image analysis have gained considerable attention. Keon et al developed a new multiscale image segmentation technique[3], i.e. the hyperstack. The conventional (single-parent) hyperstack is characterized by the fact that a voxel at one level of the hyperstack is connected to at most one (parent) voxel in next higher layer. The extension, probabilistic (multiparent) hyperstacks, is introduced, in which children are allowed to link to multiple parents, but its computational cost is very expensive. In the hyperstack segmentation method, many linkage criteria are proposed to build the most possible child-parent relationship, which demonstrates the similarity between the voxels at adjacent level efficiently. Some fuzzy clustering

approaches based on multiresolution framework were proposed recently to improve the segmentation result[4].

This paper develops a novel fuzzy segmentation framework based on multi-scale vessel filtering and the similarity between different scales. The first step of the approach is to build the feature images and relationship between child-parent scales. The fuzzy clustering combining the inter-scale and intra-scale constraints is then applied for image segmentation.

2 Multiscale Vessel Analysis

The multiscale vessel enhancement filtering was first presented in Reference [9]. The filter depends on the eigenvalues $\lambda_{\sigma,\kappa}$ ($\kappa = 1,2,3$; σ is the scale) of the Hessian Matrix of the second order image structure. The eigenvalues show the speed of intensity variation in the images. The corresponding eigenvectors express three orthonormal directions: $v_{\sigma,1}$ indicates minimum intensity variation, i.e., the direction along the vessel. $v_{\sigma,2}$ and $v_{\sigma,3}$ are orthogonal to $v_{\sigma,1}$. The ideal tubular structure in 3D images is: $|\lambda_{\sigma,1}| \approx 0, |\lambda_{\sigma,1}| \ll |\lambda_{\sigma,2}|, \lambda_{\sigma,2} \approx \lambda_{\sigma,3}$. Three basic ratios for distinguishing tubular structure and background are defined in Reference [9].

$$R_B = \frac{|\lambda_1|}{\sqrt{|\lambda_2\lambda_3|}}, R_A = \frac{|\lambda_1|}{|\lambda_2|}, S = \sqrt{\sum_{j \leq D} \lambda_j^2} \quad (1)$$

And, the vessel enhancement filter $\nu(x, \sigma)$ at location x and at scale σ is also defined. The vesselness measure provided by the filter response at different scales can obtain a final estimate of the vesselness or vessel probability:

$$\nu(x, \sigma) = \begin{cases} 0 & \text{if } \lambda_2 > 0 \text{ or } \lambda_3 > 0 \\ (1 - \exp(-\frac{R_A^2}{2\alpha^2})) \exp(-\frac{R_B^2}{2\beta^2})(1 - \exp(-\frac{S^2}{2c^2})) & \end{cases} \quad (2)$$

α, β, c are parameters, which control the sensitivity of the line filter to the measures. The filter is applied at multiple scales that span the range of expected vessel widths according to the image anatomy. The response of multi-scale filter will be maximum at the scale that approximately matches the size of the vessel to detect. Therefore, the maximum response at the matched scale is applied to obtain a final estimate of vesselness or vessel probability:

$$\nu(x) = \max_{\sigma_{min} \leq \sigma \leq \sigma_{max}} \nu(\sigma, x) \quad (3)$$

3 Nonlinear Diffusion Linking Model

In this section, we introduced the nonlinear diffusion linking model[11]. Three steps, blurring and subsampling, creating the feature images and linking, build different scales or layers in the linking model.

3.1 Blurring and Subsampling

Perona and Malik showed that a scale-space could be represented by a progression of images computed by the heat diffusion equation[10]. The pixel values at high level may be computed by successively applying diffusion equation and then subsampling. We use the P-M diffusion equation to blur the vessel image after several iterations to reduce the noise influences.

3.2 Creating the Feature Images

Now, a series of vessel intensity images at different scales (layers or levels) are created. The vesseness response (equation 3) is calculated in the layers respectively and then the feature images based on the response are used to build the linking model mentioned below.

3.3 Linking

In the linking step, the parent-child relationship between any two adjacent layers is defined. Meanwhile, the spatial relationship, between the image elements of two successive layers of the scale space, is always known. For example: in a 2^*2 subsample scale space, a pixel in higher level is the parent of four nearest image elements (children) in lower level. Here, this parent is called explicit parent, because it is exclusive. Similarly, the four children are called explicit children.

However, this relationship is ambiguous or fuzzy in a linked model. The similarity between a child image element and its possible parents is defined to describe how similar they are. Two usual similarity criteria for linking were presented [6]. The potential parent with the highest affection value is selected to be the child's parent. This affection is defined as: $L(x, y) = (\omega_1 \cdot S_I(x, y) + \omega_2 \cdot S_G(x, y)) / (\omega_1 + \omega_2)$, where ω_1 and ω_2 are weight values. S_I and S_G are the two similarity criteria[6]. x and y are a given child and potential parent respectively.

4 Multi-resolution Fuzzy Segmentation Framework

4.1 Self-similarity and Constraints of Inter- and Intra-Scale

There is self-similarity in a series of images of scale space, because all of them are the approximate representation of original image with different scales.

(1) The similarity in a scale. The similarity in a scale shows obvious clustering features, where the children belonging to identical class are similar.

(2) The similarity between two successive scales. The children are related to their parents, and the children both inherit the features of their parents, and show some new features.

In order to better describe the relations, some mathematical symbols are introduced. Let $X^{(L)} = \{X_k^{(L)} | k \in I^{(L)}\}$ be the image (or feature image of the image) in level L . $x_k^{(L)}$ is the image value or feature vector of the pixel k ,

$I^{(L)}$ is the data set. The labeled image is denoted by $l^{(L)} = \{l_k^{(L)} | k \in I^{(L)}\}$, where $l_k^{(L)} \in \{1, 2, \dots, c\}$ represents the label of the pixel k , c is the number of clustering. The multiresolution segmentation is described as: given $X^{(L+1)}$ and $l^{(L+1)}$ in the higher level $L + 1$, the optimal estimation about $l^{(L)}$ should obey the self-similarity of both inter- and intra-scale. Let $P(x_k^{(L)})$ be defined as the parent of the pixel k in level L according to the linking model, and $PS(x_k^{(L)})$ be the explicit parent of pixel k according to the spatial relationship. $P(x_k^{(L)})$ is defined as:

$$P(x_k^{(L)}) = \operatorname{argmax}\{L(x_m^{(L+1)}, x_k^{(L)})\} \quad m \in N_p(PS(x_k^{(L)})) \quad (4)$$

Where $N_p(PS(x_k^{(L)}))$ is defined as the neighbors of $PS(x_k^{(L)})$, $L(x, y)$ is the affection value between x and y described in the above section. The equation (4) is straightforward. The linked parent $P(x_k^{(L)})$ should be the pixel with maximum affection value in the neighbors of the explicit parent obtained by the spatial relationship. The neighbors decide the search volume of potential parents. For example, the 4-neighbors or 8-neighbors is usual search volume. Similarly, the linked child can be obtained from the linking relationship. Let $S(x_k^{(L+1)})$ be the most possible child of pixel $x_k^{(L+1)}$:

$$S(x_k^{(L+1)}) = \operatorname{argmax}\{L(x_k^{(L+1)}, x_m^{(L)})\} \quad m \in N_c(x_k^{(L+1)}) \quad (5)$$

Where $N_c(x_k^{(L+1)})$ denotes the explicit children of pixel $x_k^{(L+1)}$. For example, every pixel in the level $L + 1$ has four explicit children in the level L , if the 2^*2 subsample is applied in the construction of the linking model. $S(x_k^{(L+1)})$ is the child with the maximum affection value in the four children.

4.2 Multi-resolution Fuzzy Segmentation Energy Function

According to the self-similarity described above, the fuzzy distances should include two parts: (1) the fuzzy distance in a scale. (2) the fuzzy distance between two adjacent scales. The intra-scale distance is defined in the conventional FCM. The similarity between adjacent scales shows that the fuzzy clustering, the parent and its children belong to, is similar. Moreover, the clustering centers in two adjacent levels are also close. Therefore, two inter-scale fuzzy distances, i.e. $\|P(x_k^{(L)}) - v_i^{(L+1)}\|$ and $\|v_i^{(L)} - S(v_i^{(L+1)})\|$, are introduced, where $v_i^{(L)}$ is the clustering center in Level L . In figure 1, the dashed line represents the parent-child relationship based on the linking model. The first fuzzy distance shows the distance between the linked parent $P(x_k^{(L)})$ of a pixel $x_k^{(L)}$ and the corresponding clustering center $v_i^{(L+1)}$ in the higher level. The second fuzzy distance describes the distance between two corresponding clustering centers of the adjacent levels. The two distances should be minimized while the fuzzy clustering converges in a global optimal solution. The inter-scale constraint between the current level L

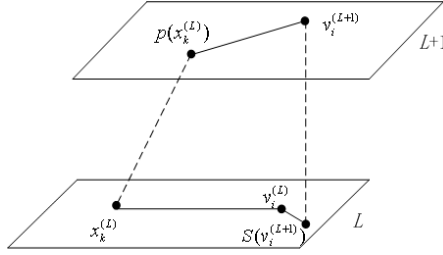


Fig. 1. The inter-scale fuzzy distance

and higher Level $L + 1$ is defined as follows:

$$J_m^{(L,L+1)}(U; V; X) = \sum_{k=1}^n \sum_{i=1}^c u_{ik}^{(L)m} [\alpha \cdot \|P(x_k^{(L)}) - v_i^{(L+1)}\|^2 + \beta \cdot \|v_i^{(L)} - S(v_i^{(L+1)})\|^2] \tag{6}$$

Where α, β are parameters, which control the sensitivity of inter-scale constraint. The membership value u_{ik} defines the grade of a feature point x_k belonging to the cluster center v_i . In most cases, m is set to be 1.5. L and $L + 1$ denote the adjacent levels. n and c are the number of feature points and fuzzy clustering respectively. According to FCM, the intra-scale constraint in Level L is defined as follows:

$$J_m^{(L)}(U; V; X) = \sum_{k=1}^n \sum_{i=1}^c u_{ik}^{(L)m} \cdot [\omega \|x_k^{(L)} - v_i^{(L)}\|^2] \tag{7}$$

Where ω is a parameter. Integrate (6) and (7), the multiresolution energy function combining the inter- and intra-constraint is defined as:

$$E(U^{(L)}, V^{(L)} | U^{(L+1)}, V^{(L+1)}) = J_m^{(L)} + J_m^{(L,L+1)} \tag{8}$$

4.3 Multiresolution Fuzzy Segmentation Algorithm

Obviously, the optimal problem in every level is to minimize the energy function described above:

$$(U^{(L)}, V^L) = \operatorname{argmin} E, \quad \sum_{i=1}^c u_{ik}^{(L)} = 1 \quad \text{for all } x_k \tag{9}$$

From Lagrange method, the iteration equations of multiresolution fuzzy clustering in level L can be obtained:

$$u_{ik}^{(L)} = \frac{(\frac{1}{H})^{\frac{1}{m-1}}}{\sum_{i=1}^c (\frac{1}{H})^{\frac{1}{m-1}}} \tag{10}$$

$$v_i^{(L)} = \frac{\sum_{k=1}^n [u_{ik}^{(L)m} \cdot (\omega \cdot x_k^{(L)} + \beta \cdot S(v_i^{(L+1)}))]}{\sum_{k=1}^n [u_{ik}^{(L)m} \cdot (\beta + \omega)]} \tag{11}$$

where $H = [\omega \|x_k^{(L)} - v_i^{(L)}\|^2 + \alpha \|P(x_k^{(L)}) - v_i^{(L+1)}\|^2 + \beta \|v_i^{(L)} - S(v_i^{(L+1)})\|^2]$. The segmentation begins from the top level, where the pre-segmentation is performed by a conventional clustering method, such as FCM. The equation (10) and (11) are the iteration equations of multiresolution fuzzy clustering in level L .

5 Experiments

We design two groups of experiments in this section. In the first group, a 185*189 synthetic image is drawn to describe the real vessel branches, where a line-like hierarchy structure represents the vessel or other line-like structures. Next we design two groups of images based on the synthetic image for simulating the real vessel images. The first group is Gaussian noise images with zero mean and different standard deviation. The standard deviations of Fig. 2(1) to Fig. 2(4) are 25,35,45,55 respectively. The images in the second group are blurred images and different Gaussian templates are applied on them. From Fig. 3(1) to Fig. 3(4), the standard deviations of templates are 2, 3, 4, 5.

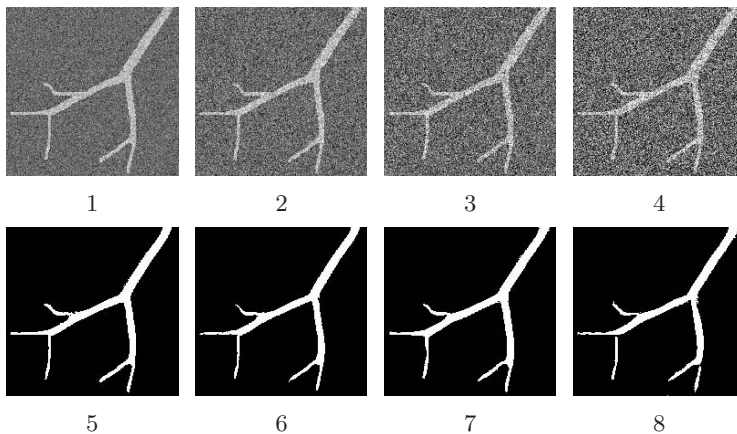


Fig. 2. Synthetic images with different noise and segmentation results

Fig2(5)-(8) and Fig3(5)-(8) are the segmentation results of the images above respectively, which demonstrates the performance of our approach. It is robust to noisy and smooth images.

In the following experiments, we present the segmentation results of 2D medical pulmonary vessel image. Meanwhile, some very small regions in the segmentation results are ignored automatically, because they may be false images or noises. Fig. 4(4) is the final result, where the vessels, especially narrow thin branches, can be extracted successfully. Moreover, many blurry and even broken branches can be captured and connected automatically.

The parameters $\alpha \in [0.5, 10]$, $\beta \in [1, 3]$ represent the confidence in the inter-scale constraints. When the noise becomes more serious, α, β should increase

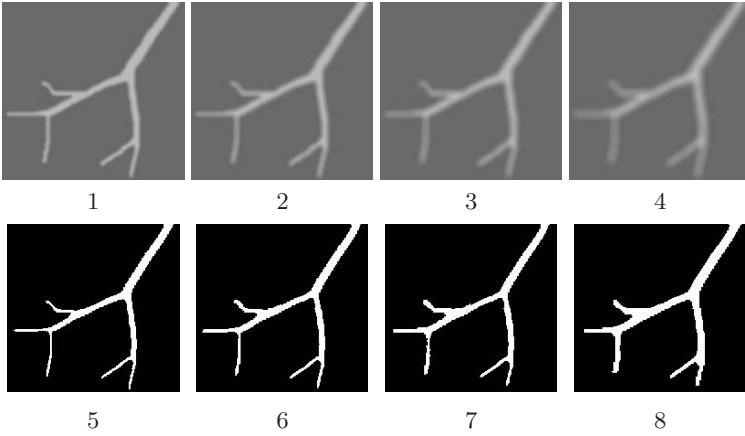


Fig. 3. Synthetic images with different gaussian smooth and segmentation results

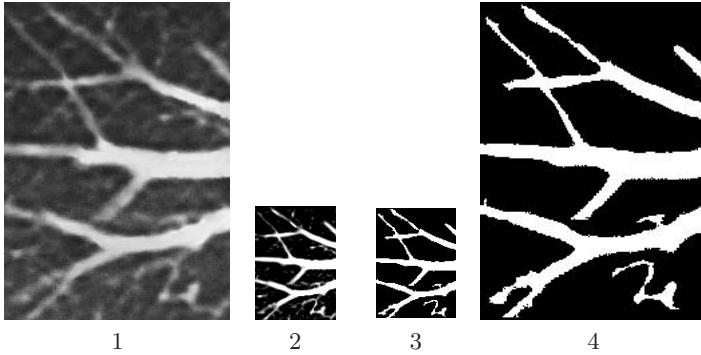


Fig. 4. The segmentation results of pulmonary vessel image. (1) original image, (2) feature image of vessel enhancement filtering in the coarse scale. (3) The segmentation result of coarse feature images. (4) The final result.

accordingly. In most cases, the value of α , β can be set as an integer, where the increment is 1. In the above experiments, we choose $\alpha \in [0.5, 1]$, $\beta = 1$, $\omega = 1$. The computation efficiency of proposed model is very desirable, which is close to conventional FCM. Every computation time of the above experiments approximates to several seconds, where our model is programmed by Matlab 7 in the computer with CPU P4-1.6GHz.

6 Conclusion

In this paper, we proposed a novel efficient multi-resolution fuzzy segmentation framework, which is based on the shape analysis of multi-scale vessel images using the multiscale vessel enhancement filter. This model is not only efficient

for the vessel segmentation, but also for other line-like structures. This proposed segmentation schema is based on the fact that the image segmentation results should be optimized simultaneously in different scales. Two fuzzy distances defined based on the constraint showed the similarity of parent-child pixels and clustering centers in successive scales. We developed a new energy function and then embedded it into the conventional fuzzy clustering. The approach is robust to noisy and low contrast vessel images, because the optimization is applied in different scale. We segmented several images including synthetic vessel and pulmonary vessel images with different noise and contrast. Segmentation results showed that the proposed approach is robust for extracting the objects.

Acknowledgement

This work was supported by the Foundation of the education Department of FuJian Province (2006F5024).

References

1. Chaudhuri, S., Chatterjee, S., Katz, N., Nelson, M., Goldbaum, M.: Detection of blood vessels in retinal images using two dimensional matched filters. *IEEE Trans. on Medical Imaging* 8(3), 263–269 (1989)
2. Thackray, B.D., Nelson, A.C.: Semiautomatic segmentation of vascular network images using a rotating structuring element (ROSE) with mathematical morphology and dual feature thresholding. *IEEE Trans. On Medical Imaging* 12(3), 385–392 (1993)
3. Koen, L.V.: Probabilistic Multiscale Image Segmentation. *IEEE Trans. Pattern Analysis and Machine Intelligence* 19(2), 109–120 (1997)
4. Sokratis, M.: Segmentation of Color Images Using Multiscale Clustering and Graph Theoretic Region Synthesis. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans* 35(2), 224–238 (2005)
5. Nikos, P.: Geodesic active regions: A new framework to deal with frame partition problems in computer vision. *Journal of Visual Communication and Image Representation* 13, 249–268 (2002)
6. Yezzi Jr., A., Andy, T., Alan, W.: A Fully Global Approach to Image Segmentation via Coupled Curve Evolution Equations. *Journal of Visual Communication and Image Representation* 13, 195–216 (2002)
7. Pascal, M., Philippe, R., Francois, G., Prederic, G.: Influence of the Noise Model on Level Set Active Contour Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26(6), 799–803 (2004)
8. Ali, G., Raphael, C.: A new fast level set method. In: *Proc. of the 6th Signal Processing Symposium*, pp. 9–11 (2004)
9. Frangi, A.F., Niessen, W.J., Vincken, K.L., Viergever, M.A.: Multiscale vessel enhancement filtering. *LNCS*, vol. 1946, pp. 130–137. Springer, Heidelberg (1998)
10. Perona, P., Malik, J.: Scale-space and Edge Detection using Anisotropic Diffusion. *IEEE Transaction On Pattern Anal. and Mach. Intell* 12(6), 629–639 (1990)
11. Yu, G.: A Novel Fuzzy Segmentation Approach for Brain MRI. In: Fischer, K., Timm, I.J., André, E., Zhong, N. (eds.) *MATES 2006. LNCS (LNAI)*, vol. 4196, pp. 887–896. Springer, Heidelberg (2006)

Building a Spanish MMTx by Using Automatic Translation and Biomedical Ontologies

Francisco Carrero¹, José Carlos Cortizo^{1,2}, and José María Gómez³

¹ Universidad Europea de Madrid, C/Tajo s/n, Villaviciosa de Odón, 28670, Madrid, Spain
{francisco.carrero, josecarlos.cortizo}@uem.es

² Artificial Intelligence & Network Solutions S.L.
jccp@ainetsolutions.com

<http://www.ainetsolutions.com/jccp>

³ Departamento de I+D, Optenet, Parque Empresarial Alvia, 28230, Las Rozas, Madrid, Spain
jgomez@optenet.com

Abstract. The use of domain ontologies is becoming increasingly popular in Medical Natural Language Processing Systems. A wide variety of knowledge bases in multiple languages has been integrated into the Unified Medical Language System (UMLS) to create a huge knowledge source that can be accessed with diverse lexical tools. MetaMap (and its java version MMTx) is a tool that allows extracting medical concepts from free text, but currently there not exists a Spanish version. Our ongoing research is centered on the application of biomedical concepts to cross-lingual text classification, what makes it necessary to have a Spanish MMTx available. We have combined automatic translation techniques with biomedical ontologies and the existing English MMTx to produce a Spanish version of MMTx. We have evaluated different approaches and applied several types of evaluation according to different concept representations for text classification. Our results prove that the use of existing translation tools such as Google Translate produce translations with a high similarity to original texts in terms of extracted concepts.

Keywords: Semantic techniques, data pre and post processing, information filtering, recommender systems.

1 Introduction

The volume of published biomedical information is increasing every year. The proliferation of online sources such as scientific repositories, clinical records databases, knowledge databases, etc., has produced an information overload that surpass the amount of information that researchers can cope with. This scenario makes it necessary to develop tools that help access and visualization of specific information useful for biomedical researchers. Pubmed¹, a service of the U.S. National Library of Medicine, constitutes an example of a huge source of biomedical information, since it includes over 17 million citations from multiple life science journals, among which stand out Medline [1].

¹ <http://www.ncbi.nlm.nih.gov/pubmed/>

Several attempts to develop common biomedical terminologies that help improving interoperability between medical resources have appeared within the last few years. However, it has not been until the appearance of UMLS (Unified Medical Language System) [3] that there is a unified way to access multiple and complementary resources. UMLS includes more than 60 families of controlled vocabularies and resources such as SNOMED-CT, MeSH, ICD-10 or Gene Ontology. This knowledge has proved useful for many applications including decision support systems, management of patient records, information retrieval and data mining.

The MetaMap system [2] is an online application developed at the National Library of Medicine (NLM) that allows mapping text to UMLS Metathesaurus concepts, which is very useful for interoperability among different languages and systems within the biomedical domain. MetaMap Transfer (MMTx) is a Java program that makes MetaMap available to biomedical researchers in controlled, configurable environment. Currently there is no Spanish version of MetaMap, which difficult the use of UMLS Metathesaurus to extract concepts from Spanish biomedical texts. Developing a Spanish version of MetaMap would be a huge task, since there has been a lot of work supporting the English version for the last sixteen years.

Our ongoing research is mainly focused on using biomedical concepts for cross-lingual text classification [4]. In this context the use of concepts instead of bag of words representation allows us to face text classification tasks abstracting from the language. In this paper we evaluate the possibility of combining automatic translation techniques with the use of biomedical ontologies to produce an English text that can be processed by MMTx.

1.1 Project Description

MIRCAT (Multilingual Information Retrieval based on Concepts and Automated Translation) is a cross-lingual system to retrieve biomedical documents significantly related to medical records. Given a query in Spanish, it retrieves a list of medical records ordered by relevance in two steps: 1) the query is expanded using concepts included in a biomedical ontology; 2) medical records are ranked using a representation based on biomedical concepts. Then, the user can choose a record and the system will retrieve several lists of ranked documents as follows: a) Spanish news; b) English news; c) Spanish article abstracts; and d) English article abstracts. This last step uses concepts to rank the documents against the selected medical record.

Throughout all the phases we need to obtain a semantic document representation, which makes it definitely crucial to use an accurate system to extract concepts from text. Since we are mainly working with UMLS, we face the issue that currently there is only an English version of MetaMap and MMTx. The development of an equivalent tool in Spanish would require a huge amount of work and specific knowledge and, although it would be a very valuable task, we wonder if it is really a must.

The key point for us at current stage is to evaluate the need to develop a Spanish version of MMTx, against the possibility of using automatic translation systems (such as Google Translator or Systran) to obtain an English representation for a Spanish text, and then to apply MMTx to English text and obtain a semantic representation that should include (almost) the same concepts as in Spanish.

2 Related Work

The most widely used text representation in text classification tasks such as Information Retrieval (IR) or Text Categorization (TC) has been, by far, the bag of words model [12]. In this representation, a document is represented as vector of terms and associated weights. Terms are usually stemmed words, and weights are computed as a function of their occurrences in documents and the whole text collection, like TF.IDF weights. This representation does not capture the full meaning of texts, but it is enough to build reasonably effective text classifiers.

However, there have been several attempts to design text representations that better capture the semantics of documents. These approaches rely on the emergence of wide-coverage semantic resources like WordNet. For instance, some authors have demonstrated that using WordNet concepts (synsets) instead of, or added to, words can improve Information Retrieval [9] and Text Categorization [8].

A major point is that concepts can be language-independent (as in EuroWordNet), what allows full cross-language retrieval and categorization [10]. However, concept based representations (concept indexing) are doomed with the limited effectiveness of current free text Word Sense Disambiguation (WSD) approaches. The effectiveness of an average WSD system rarely exceeds 60% on ambiguous words (see e.g. Seneseval [13] results) on running text, a level that is hardly reached on short texts like search engine queries. On the other side, previous works have demonstrated that the effectiveness of text classification can be improved even in the presence of an important percentage of disambiguation errors.

A promising issue is that there are high quality semantic resources in the domain of biomedicine, like the Unified Medical Language System (UMLS) or SNOMED. These resources have been successfully used in several text classification tasks. For instance, [14] reports good results when using UMLS concepts for concept indexing in the European Project MUCHMORE. Also, [11] presents the MorphoSaurus system, which makes use of UMLS for concept indexing in cross-language retrieval, in comparison with query translation, with results that support concept indexing.

Regarding translation, a full report of the current state of the art is beyond the scope of this paper. Instead, let us remark that the system we employ, Google statistical translator, has top performed in the most recent NIST Open Machine Translation Competition (2006). The strength of this translation tool relies on the huge amount of data it makes use for computing the statistical metrics of its model.

3 Spanish MMTx

We have developed two versions of Spanish MMTx: A first simple approach uses Google Translator to obtain an English version of the text and then applies English MMTx to extract concepts. This approach, ignoring the quality of general translation, presents some important mistakes when translating some technical biomedical terms, keeping them in Spanish.

The second approach delegates to Google Translator to obtain the general translation, but uses a custom UMLS ontology mapper to translate biomedical terms. The first version of the custom UMLS ontology mapper has been created building a sub-ontology of

UMLS by using only the “isa” relation. Then, for each of the concepts included, all Spanish and English string representations have been stored. Considering this mapper, this second approach involves the following steps:

1. Search the original Spanish text and substitute each of the found concepts by its concept ID. In case of ambiguity, the chosen concept is the one with higher level in the ontology.
2. Send the text from the first step to Google Translator, retrieving an English version with the general translation.
3. Search the English version and replace the concept IDs with a string representation. If there are several representations, we chose to use the shortest one.
4. Use the English MMTx to extract the concepts.

4 Experiments

In the previous section we stated our hypothesis: using automated translation combined with the use of domain ontologies and MMTx, we could avoid the need of a specific Spanish MetaMap. To test the validity of this hypothesis, we need to compare the concepts extracted by MMTx from English texts to the concepts extracted by MMTx from Spanish texts previously translated to English.

4.1 Description an Goals

For testing our hypothesis, we needed a corpus of biomedical documents in both languages: Spanish and English. MedLine Plus stores health-related news articles extracted from Reuters Health and HealthDay. All these news articles are tagged with a set of related MedLine Plus pages, which can be considered as topics or categories (there are over 750 different diseases or conditions treated as topics). MedLine Plus contains medical information in English and also some of the contents in Spanish. We developed a spider that, once a month, downloaded all the English and Spanish news articles and checked the correspondence among news. From over 2000 news downloaded since December 2007, we were able to detect 600 news articles available in both languages and we built the collection using those items.

Our approach in this paper evaluates concept based document representations for no particular text classification task. The main goal is to establish whether our approach could produce benefits to any text task or if it should not be considered.

From our original bilingual collection of news articles, we have generated 3 different collections:

- ENG: Containing the original English documents.
- ENG_TRANS: Containing the Spanish documents automatically translated to English using Google Translator.
- ENG_UNMKD: Containing the translations to English by means of Google Translator and domain ontologies (UMLS), as described in section 3.

4.2 Possible Documents Representation

There are two important considerations from the MMTx representation. A string like C0331964 represents a concept. Some phrases are represented by a conjunction of concepts, which is represented by several strings connected by ‘|’. There are some phrases that appear several times, meaning that there are ambiguities or different possible concepts or combination of concepts that represents that phrase.

We should translate the MMTx representation to a representation containing a list of concepts. Paying attention to the previous considerations of the MMTx representation, we should deal with compound concepts and with ambiguities. We have developed 4 possible data representations according to this: A1, A2, B1, B2.

- Document representations starting with an ‘A’ (A1 and A2) uses compound concepts. That means that a compound concept like C0205388|C0439227|C0439228 would be treated as a simple one like C0331964.
- In document representations starting with a ‘B’ (B1 and B2) compound concepts are divided into simple concepts as indexing units. A concept like C0205388|C0439227|C0439228 is transformed into 3 different concepts (C0205388, C0439227 and C0439228).
- Document representations ending with ‘1’ (A1 and B1) resolve the ambiguity by adding all the concepts contained in all the possible interpretations of the phrase.
- Document representations ending with ‘2’ (A2 and B2) ignore the ambiguities by choosing the first possibility for each phrase.
- We have also tested a word based representation as baseline.

A1 document representation is more complex and nearer to the human understanding and B2 document representation is the simplest one and nearer to the standard machine representation for text mining tasks. More complex document representation generates more different concepts. Table 1 shows the number of global concepts for each document representation.

Data representations containing a lot of features do not usually perform very well in text tasks, especially in text classification, as many classifiers degrade in prediction accuracy when faced with many irrelevant features or redundant/correlated ones [5]. The explanation to this phenomenon may be found in the “curse of dimensionality”, which refers to the exponential growth of the number of instances needed to describe the data as a function of dimensionality (number of attributes). Zipf’s Law can be used to solve this problem without facing any concrete task, by filtering the features appearing in more than M% of the documents and the ones appearing in less than N% of the documents. We have filtered the concepts according to this, with M=10% and N=1%. The global number of concepts after this filtering process is shown in Table 1.

Table 1. Different concepts for each document representation and number of concepts after filtering

Document representation	Total	Filtered
A1	45.280	2.368
A2	21.257	1.415
B1	9.990	2.293
B2	8.148	1.653
Word	15.966	2.665

Table 2. Average similarities between document representations generated from translated texts and the representations generated from the original English texts

Document representation	TRANS	UNMKD
A1	56.86±8.37	54.31±7.90
A1+Zipf	65.87±11.11	63.23±10.99
A2	60.79±6.78	58.07±6.40
A2+Zipf	65.80±9.56	62.94±9.51
B1	79.42±6.43	76.55±5.54
B1+Zipf	77.63±8.85	75.00±8.56
B2	78.38±6.21	74.76±5.38
B2+Zipf	76.38±8.53	73.59±8.18
Word	75.11±6.13	72.69±8.09
Word+Zipf	73.45±5.21	70.30±7.55

4.3 Results

Table 2 resumes the results of these experiments. We have computed the similarity between the original ENG documents and the translated ones (ENG_TRANS and ENG_UNMKD) for each possible representation. Then, we have calculated the average value and standard deviation for the 600 news items in the global collection.

5 Discussion of the Results

Considering the four representations described above, the worst results in terms of similarity are achieved with the most complex and near-to-humans representation (A1). On the other side, B1 is a less complex and near-to-humans representation, and produces the best results of the series. This proves that our model seems to be more suitable for tasks that manage the concepts on a plain bag-of-concepts way.

The use of Zipf's law improves the results within the A representations, while makes the values obtained for B get worse. The reason for A may possibly be that this representation produces too many different concepts, because some of them are made up of combinations of simpler ones and many of them appear few times on the text. Since we keep only the most relevant concepts, it seems to eliminate some of the concepts that make the difference for each pair of documents. The loss of precision obtained with representation B may come from the fact that the initial number of concepts is already low.

Relating to the difference between the results when applying simple or complex custom UMLS concepts mapper, it is clear that the complex one currently does not improve the translation over the simple one, although the difference isn't too high. It may be to some extent due to several limitations on the translator that are described below but, however, there are enough things to improve on the mapper.

It is interesting to see how simple conceptual representations (B1 and B2) obtains better similarity values than baseline word-based representations. Also, we consider that values of 79,42% for the simple mapper and 76,55% for the complex one are promising enough to continue with our research on improving the models. Specially, we find that there is a broad field to improve the complex UMLS ontology mapper.

6 Conclusions and Future Work

To date, there isn't an effective tool to extract UMLS concepts from Spanish texts. Our experiments on creating a Spanish MMTx combining existing English MMTx and automatic translators have shown to be promising for tasks such as Text Categorization and Information Retrieval as the concept based representation of translated text does not vary much from the concept based representation of English documents. However, it is out of the scope to evaluate the correctness and quality of translation. Of course, a specific Spanish MMTx will always be more accurate than this model, but the key point is to consider if such a huge task would improve further results in TC and IR.

Our next step will be to apply the Spanish MMTx to diverse text mining tasks, like Text Categorization or Information Retrieval. Testing the documents representations evaluated in this paper on real text tasks, will allow us to conclude if there is any need to build a Spanish MMTx from scratch.

We will try modifying our custom UMLS ontology mapper, using more semantic relations and keeping only those concepts that can be considered to belong to the biomedical domain. From a more practical point of view, we are currently developing more sophisticated techniques to retrieve similar documents based on conceptual representations using probabilistic models, machine learning algorithms [7] and feature selection techniques [6].

Acknowledgments

This work has been partially funded by the Spanish Ministry of Education and Science and the European Union from the ERDF (TIN2005-08988-C02), and the Spanish Ministry of Industry as part of the PROFIT program (FIT-350300-2007-75).

References

1. MEDLINE Factsheet,
<http://www.nlm.nih.gov/pubs/factsheets/medline.html>
2. Aronson, A.R.: Effective mapping of biomedical text to the UMLS Metathesaurus. In: Proceedings of the American Medical Informatics Association Symp., pp. 17–21 (2001)

3. Bodenreider, O.: The Unified Medical Language System (UMLS): integrating biomedical terminology. *Nucleic Acids Research* 2004 32, D267–D270 (2004)
4. Carrero García, F., et al.: Attribute Analysis in Biomedical Text Classification. In: Second BioCreAtIvE Challenge Workshop: Critical Assessment of Information Extraction in Molecular Biology, Spanish Nacional Cancer Research Centre (CNIO), Madrid, SPAIN (2007)
5. Cortizo, J.C., Giraldez, I.: Discovering Data Dependencies in Web Content Mining. In: Proceedings of the IADIS International Conference WWW/Internet 2004, Madrid, Spain, October 6-9, 2004, pp. 881–884 (2004)
6. Cortizo, J.C., Giraldez, I., Gaya, M.C.: Wrapping the Naïve Bayes Classifier to Relax the Effect of Dependences. In: Yin, H., Tino, P., Corchado, E., Byrne, W., Yao, X. (eds.) IDEAL 2007. LNCS, vol. 4881, pp. 229–239. Springer, Heidelberg (2007)
7. Gaya, M.C., Giraldez, I., Cortizo, J.C.: Uso de algoritmos evolutivos para la fusion de teorías en minería de datos distribuida. In: Actas de la XII Conferencia de la Asociación Española para la Inteligencia Artificial – CAEPIA/TTIA 2007, vol. 2, pp. 121–130 (2007)
8. Gómez Hidalgo, J.M., et al.: Concept Indexing for Automated Text Categorization. In: Meziane, F., Métais, E. (eds.) NLDB 2004. LNCS, vol. 3136, pp. 195–206. Springer, Heidelberg (2004)
9. Gonzalo, J., et al.: Indexing with WordNet synsets can improve Text Retrieval. In: Proceedings of the COLING/ACL 1998 Workshop on Usage of WordNet for Natural Language Processing, Montreal (1998)
10. Gonzalo, J., et al.: Applying EuroWordNet to Cross-Language Text Retrieval. *Computers and the Humanities* 32, 2–3, 185–207 (1998)
11. Marko, K., Schulz, S., Hahn, U.: MorphoSaurus—design and evaluation of an interlingua-based, cross-language document retrieval engine for the medical domain. *Methods of Information in Medicine* 44(4), 537–545 (2005)
12. Sebastiani, F.: Machine Learning in Automated Text Categorization. *ACM Computing Surveys* 34(1), 1–47 (2002)
13. Snyder, B., Palmer, M.: The English all words task. In: SENSEVAL-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text (2004)
14. Volk, M., et al.: Semantic annotation for concept-based cross-language medical information retrieval. *International Journal of Medical Informatics* 67(1-3), 97–112 (2002)

Compensation for Speed-of-Processing Effects in EEG-Data Analysis

Matthias Ihrke^{1,2}, Hecke Schrobsdorff^{1,2}, and J. Michael Herrmann^{1,3}

¹ Bernstein Center for Computational Neuroscience Göttingen

² MPI for Dynamics and Self-Organization, Bunsenstr. 10, 37073 Göttingen, Germany

³ Institute for Perception, Action and Behaviour, University of Edinburgh
Informatics Forum, 10 Crichton Street, Edinburgh, EH8 9AB, U.K.

ihrke@nld.ds.mpg.de, hecke@nld.ds.mpg.de,
michael.herrmann@ed.ac.uk

Abstract. We study averaging schemes that are specifically adapted to the analysis of electroencephalographic data for the purpose of interpreting temporal information from single trials. We find that a natural assumption about processing speed in the subjects yields a complex but nevertheless robust algorithm for the analysis of electrophysiological data.

1 Introduction

In electroencephalographic (EEG) data noise levels of -25dB are not uncommon [1], for electromyography (EMG) or functional magnetic resonance imaging (fMRI) the situation is similar. The arising problem of the recovery of relevant information from such data has been dealt with extensively [2,3,4]. It seems reasonable to exploit intrinsic structures in the data, i.e. to identify patterns in the data that reoccur under specific conditions, e.g. at the onset of a stimulus or in relation with other events in the course of the experiment.

A straight-forward solution consists in averaging single-trial *event related potentials* (ERPs) in order to obtain an averaged ERP (AERP) that is expected to be comparable across different experimental setups [3]. The reliability of the AERP allows for the identification of characteristic features of the time course of the signal such as the latency and amplitude of major minima and maxima, which may be used iteratively to further improve the process of averaging. We will discuss several algorithms that are not only theoretically justified, but have proven useful also in an experimental project [5,6]. Systematic changes in the AERP components between different experimental conditions are consistent with the hypothesis that ERP components do reflect stages of information processing in the brain. In this interpretation, the idealized noise-free ERP represents the signal of interest and variability across trials is merely noise.

The data that gave rise to this studies has been obtained in a series of EEG experiments. A subject was engaged in a priming task (cf. [5] for more details on the priming effect) and supposed to respond to a stimulus. The stimulus configuration presumably triggered various modes of internal processing in the subject, cf. Fig. 1. Some of the behavioral effects turned out to be fragile and require a large number of trials to

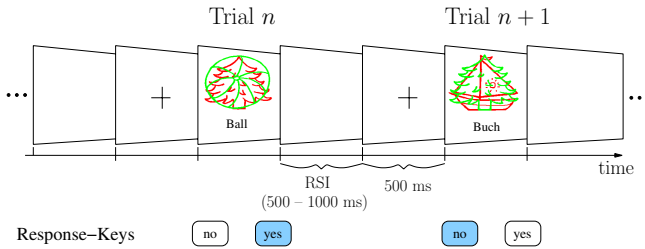


Fig. 1. Time course of a trial of the psychophysical experiment. Subjects are asked to identify the green (light gray) stimulus in presence of a red (dark gray) distractor. If the target matches the written word then the correct response is “yes”. Stimulus presentations are interleaved with response-to-stimulus intervals (RSI) and fixation phases. Depending on which stimuli from the previous trial are repeated, reaction time can be increased or decreased.

become significant. The underlying information processing mechanisms are studied by simultaneous EEG recordings which we assumed to obey the following conditions.

1. The EEG signal contains the relevant components of the neural activity.
2. Task-specific activations form a significant fraction of the EEG signal.
3. The brain solves similar tasks in a similar way.

The *signal* can now be defined as a minimal variance curve within the data set obtained for many repetitions of the same task. The axioms imply that variations due to external conditions should be excluded and that the external conditions and even the state of the subject should be kept as constant as possible for all trials. Yet, data mining techniques reveal that for comparable data only a fraction of 60% of the pooled epochs contribute to the AERP waveform while the other 40% just increase the variance [7].

Thus, it cannot be decided unambiguously whether the variability of the ERPs is caused by the stochastic nature of the underlying neural dynamics or by the application of different strategies to the task. A plot of the single-trial ERPs and the AERP, see Fig. 3 points already to a basic problem: Simple averaging will deteriorate in particular late components of the ERP (such as the *Late Positive Complex*) which which make the interpretation of these components difficult.

2 Models for Event-Related Potentials

The signal-to-noise ratio (SNR) of EEG data is typically enhanced by combining data epochs that are supposed to contain a certain signal component as a pointwise average

$$\langle s_i(t) \rangle_i = \frac{1}{N} \sum_{i=1}^N s_i(t) \quad i = 1, \dots, N. \tag{1}$$

Here $s_i(t)$ is the measurement of the i th trial, $1, \dots, n$, at time t . The *signal-plus-noise* (SPN) model [8] or *fixed-latency* model [9] underlying this average assumes that (i) signal and noise add linearly, (ii) the signal is identical in all trials, and (iii) noise is a zero-mean random process drawn independently for each trial.

Assuming additive zero-mean noise, i.e. $\langle \epsilon(t) \rangle = 0 \forall t$, we can represent the data by $s_i(t) = u(t) + \epsilon_i(t)$, where $u(t)$ denotes the signal that is to be recovered from $s(t)$. Under the above conditions the pointwise average is an unbiased and optimal estimate in the mean-square error sense. It has been argued on theoretical grounds, that an improvement beyond pointwise averaging is not possible if no *a priori* knowledge about the characteristics of signal and noise is given [10]. However, the requirement of a smooth temporal structure of the data [9] may already serve as a prior that may indeed lead to an improvement.

An argument against the stationarity of u comes from the analysis of the residuals ζ_i^{avg} obtained by subtracting the mean from the raw data $\zeta_i^{\text{avg}}(t) = s_i(t) - \langle s_i(t) \rangle_i$. The fact that the repetition of a task is typically accompanied by coherent on-going neural activity [11] can be analyzed as follows. Given the SPN model, ζ_i^{avg} should not contain any event-related modulation because the noise is assumed to be independent and identically distributed. Therefore statistical coherence measures such as the *autocorrelations*

$$(\zeta_i^{\text{avg}} \star \zeta_i^{\text{avg}})(\tau) = \int \zeta_i^{\text{avg}}(t)\zeta_i^{\text{avg}}(t + \tau) dt \tag{2}$$

and the *spectral densities*

$$\text{PSD}(\zeta_i^{\text{avg}}) = \mathcal{F}\{\zeta_i^{\text{avg}} \star \zeta_i^{\text{avg}}\} \tag{3}$$

computed on ζ_i^{avg} should not show any event-related modulation (i.e. a flat spectrum and cross correlations that behave like a δ -function at zero are to be expected). Empirical evidence shows that these assumptions are violated for real data [11, 8], see Fig. 2.

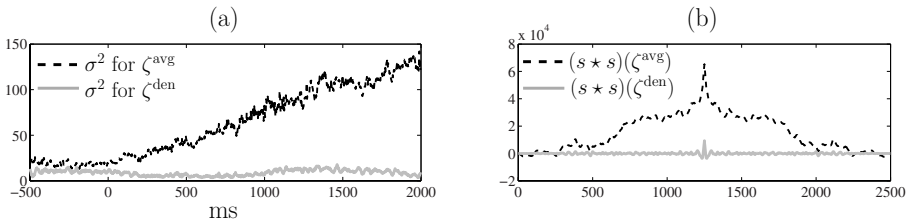


Fig. 2. Coherence measures computed on residuals ζ^{avg} . (a) The variance σ^2 over trials shows event-related modulation for the residuals after subtracting the average. Given the SPN model, we expect a flat curve as obtained from computing σ^2 on the single-trial denoised residuals ζ_i^{den} . (b) Crosscorrelation computed on the residuals for a sample trial. Again, unexpected (from SPN) correlations show up for ζ_i^{avg} whereas the function approximates a δ -function for the denoised single-trial residuals.

Extending the SPN model, the *variable latency model* (VLM) [2] introduces a trial-dependent scaling factor α_i and a time lag τ_i

$$s_i(t) = \alpha_i u(t + \tau_i) + \epsilon_i(t). \tag{4}$$

One possibility to obtain the τ_i is the maximization of the crosscorrelation $\tau_i = \arg \max_t (\langle s_i \rangle_i \star s_i)(t)$ between the data and the pointwise average $\langle s_i \rangle_i$. After this

transformation, the data can be interpreted by the SPN model. However, in an empirical evaluation of analytically derived predictions of this model, patterns that were not consistent with the predictions were found [8]. We were hence led to reconsider the fact that the intertrial variability of the evoked potential can go beyond the simple time shift [12]. A more general model for order-preserving time warping is given by

$$s_i(t) = \alpha_i(t)u(\phi_i^{-1}(t)) + \epsilon(t), \quad (5)$$

where ϕ_i are monotonous functions that map the time scale of the individual trials to that of a template v (i.e. $\|u_v(t) - u_i(\phi_i(t))\|$ is minimal). The functions $\alpha_i > 0$ determine the local scaling of the curve. The advantage of this *variable-components-plus-noise* model (VCPN) is illustrated by Fig. 3. The VCPN model (5) models temporal variations

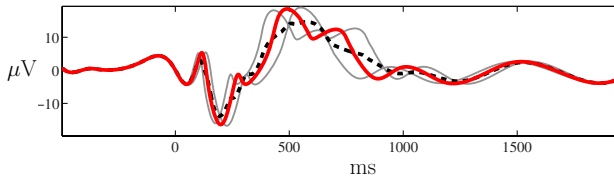


Fig. 3. “Smearing” of components in the simple average due to temporal variance. Two input signals (gray curves) were simulated according to the model in Eq. 5. An averaging procedure incorporating temporal variance would produce a curve similar to the red (solid) one.

of the individual signals in addition to differences in scale and is thus able to identify systematic distortions due to single-trial fluctuations that otherwise are averaged out. The functions ϕ_i require some regularization that we achieve by crossvalidation.

We use two datasets, one containing artificial data the other one consisting of real EEG data. The real data reported in this chapter were obtained in a study featuring a picture-identification task (for details cf. [5, 6]), see Fig. 1. The artificial data was generated according to the VCPN model introduced in Eq. 5 (Sect. 2).

3 Dynamic Time Warping

Unsupervised classification techniques can help to identify ERPs that were generated by distinct processing mechanism in the brain. The temporal variance introduced in this way, however, is not resolved by the VCPN model (5). Averaging techniques should, therefore, be applied selectively to trials within distinct clusters. *Dynamic time warping* (Fig. 5) will be shown to provide a distance measure that directly implements the assumptions on the relevant features.

Selective averaging schemes use only specific episodes for averaging [13] in order to exclude artefacts such as muscular activity. In order to reduce visual inspection of the data it is possible to assist the selection process by clustering [14]. An inherent problem of any clustering algorithm is the decision about a sensible number of clusters. If a specific experimental design allows a theory-driven estimate of that number it is of course

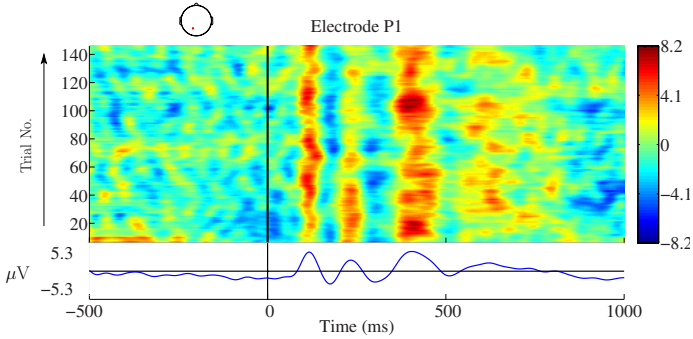


Fig. 4. Real data ERPs for 150 trials (color coded) and their average (lower part). With growing number of trials (i.e. time spent in the experiment) the shape of the ERP is subject to changes. E.g. while the P2 is very pronounced for the first 50 trials, it's amplitude is decreased later. The N2 amplitude decreases over time and is missing completely in some trials. A general shift in the pattern with growing number of trial is observable (P2/N4 amplitude).

to be preferred. Otherwise, e.g. strategies based on within-cluster scatter coefficients can be applied (e.g. [15]). Further improvement is possible by exploiting spatial relations among the electrodes, although this somewhat reduces the spatial resolution. ERP onset latencies are considered by variable-latency averaging [2] similarly as in VLM [4]. Adjacent response-overlap averaging (ADJAR) [16] aims at removing temporal jitter arising from the response time variations of the preceding trial although this scheme seems restricted to short RSIs.

Finally, *dynamic time warping* (DTW), which has been first used in speech analysis [17], appears to present a more suitable approach to EEG analysis. DTW tries to align a trial time course to a template, see Fig. 6b). First, a pointwise dissimilarity measure between two signals s_1, s_2 is defined, e.g.

$$d(s_1(t_1), s_2(t_2)) := |\tilde{s}_1(t_1) - \tilde{s}_2(t_2)| + |\tilde{s}_1'(t_1) - \tilde{s}_2'(t_2)|, \quad (6)$$

where $\tilde{s}(t) := (s(t) - \langle s(t) \rangle_t) \langle s(t)^2 \rangle_t^{-1/2}$ is the normalized signal and s' the first derivative of s . The distance (6) gives rise to derivative DTW [18] because it is based on amplitude and slope of the signal.

An optimal map is determined by a path p_i that satisfies recursively

$$\text{if } p_i = (j, k) \text{ then } p_{i+1} \in \{(j + 1, k), (j, k + 1), (j + 1, k + 1)\} \quad (7)$$

and minimizes the sum of the selected elements d_{jk} of the dissimilarity matrix [3]

$$\mathbf{d}_{jk} = d(s_1(j), s_2(k)). \quad (8)$$

This path can be found by backtracking through the *cumulated cost matrix*

$$\mathbf{D}_{jk} = \mathbf{d}_{jk} + \min \{ \mathbf{D}_{j,k-1}, \mathbf{D}_{j-1,k}, \mathbf{D}_{j-1,k-1} \}, \quad (9)$$

i.e. via the minimum of the downward, right, and down-right neighbors, from $\mathbf{D}_{J,K}$ to $\mathbf{D}_{1,1}$. The final element $\mathbf{D}_{J,K}$ constitutes a measure for the dissimilarity of the two

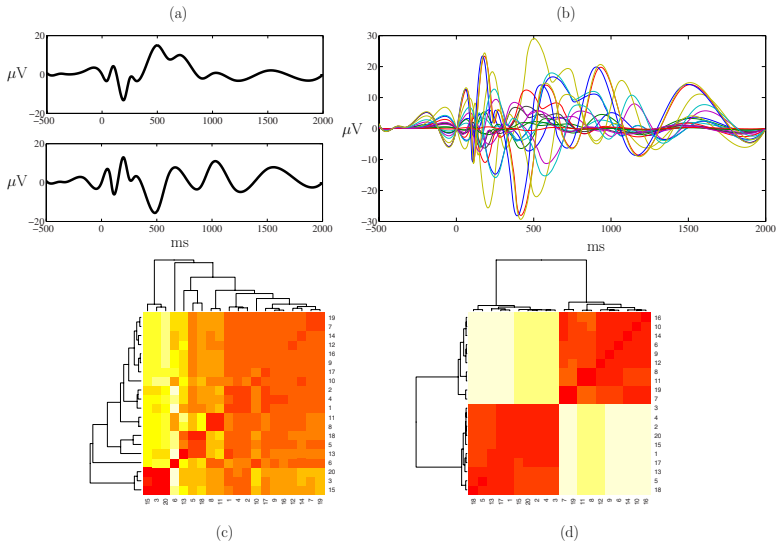


Fig. 5. Cluster analysis of denoised single-trial ERP data. (a) Two template trials were used to derive the single-trial instances in (b) according to Eq. 5. (c) Heat map based on Euclidean distances, (d) same for DTW (Sect. 3). While the DTW metric correctly classifies all trials to be generated by one of the templates in (a), the Euclidean fails to do so in several instances.

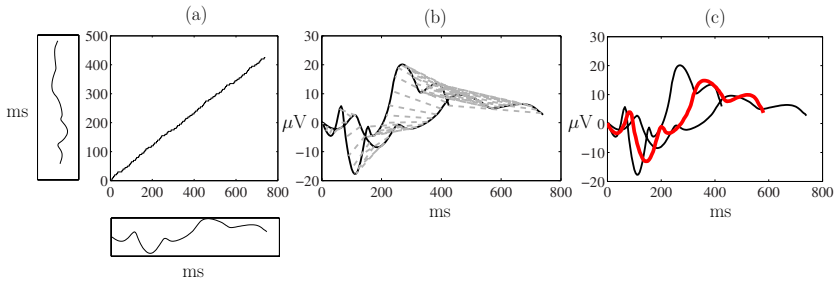


Fig. 6. Dynamic time warping. (a) Optimal path p_i through the cost-matrix D_{jk} for two signals (black curves) from (b) and (c). (b) Illustration of DTW matching corresponding points in s_1 and s_2 . (c) An average produced by ADTW (red).

curves based on their overall shape. Once this path is available, it is easy to average the curves (called *averaging dynamic time warping*, ADTW) to reduce both temporal and scale variance (see Fig. 6) by setting

$$D^{\text{ADTW}}\{s_1, s_2\}(t) = (s_1(j) + s_2(k))/2, \tag{10}$$

where $(j, k) = p_t$ as introduced in Eq. 7 and $t = 1, \dots, J + K$.

For N trials, a straightforward solution proposed in [3] is to simply combine pairs of single-trial ERPs using ADTW. In a next step, pairs of the results from this combination can be averaged again and the entire process iterated until only one average is left. Starting from Eq. 10, we proceed recursively, namely

$$D^{\text{ADTW}}\{s_1, \dots, s_{2N}\}(t) = D^{\text{ADTW}}\{D^{\text{ADTW}}\{s_1, \dots, s_N\}, D^{\text{ADTW}}\{s_{N+1}, \dots, s_{2N}\}\}(t). \quad (11)$$

It is further possible to introduce constraints on the DTW method that penalize path deviations [3] from the main diagonal, thereby reducing the bias on the cost of an increased variance. Before applying the DTW algorithm, it should be ensured that trials are sufficiently similar to each other, e.g. by applying time warping only within distinct clusters. In the next section, we propose to apply external time markers that can act as an objective reference for trial matching and advanced averaging.

4 Enhancing Averaging by Integrating Time Markers

An extension of DTW incorporates information about latency variability by hierarchically choosing pairs of similar trials before averaging. The cumulated path coefficient obtained from DTW is used as a measure for the dissimilarity of two time courses, cf. Fig. 6. After selecting the minimum element from the matrix $\Delta_{ij} = \text{DTW}\{s_i, s_j\}$ by $(i, j) = \arg \min_{(j,k)} \Delta_{jk}$, the minimal-dissimilarity trials are combined and row i and column j are removed from Δ_{jk} . This procedure is iterated until the matrix is empty. The complete process is repeated with about half the number of pairwise averaged trials. The entire tree-like process is continued until all trials are merged. The scheme is called *pyramidal ADTW* (PADTW) because the subdivision scheme of the set of trials. For realistic data, the procedure performs substantially better than DTW [6].

Some experimental setups suggest an alignment of the data w.r.t. response markers, i.e. instead of stimulus-locked now response-locked ERPs are used. In Ref. [4] it was proposed to stretch or compress the single-trial signals in order to match the average reaction time by moving the sampling points in time according to a temporal low-order power law. We employ instead a more flexible approach that integrates prior knowledge about an arbitrary number of markers by applying ADTW separately to the segments and concatenating them within the PADTW algorithm. However, this approach is equivalent to calculating the pointwise dissimilarity matrix d_{jk} from equation (6) for trial j and k on the complete dataset and manipulate this matrix before continuing with the steps outlined in the PADTW algorithm. In the manipulated matrix, the fields corresponding to an event in both trials are set to zero (minimal dissimilarity). This instructs the algorithm that the points of the two signals match and forces the DTW-path to lead through these points. It follows naturally that not only two but arbitrarily many time-markers can be integrated to guide the formation of the average. This approach has been used in Ref. [6], where the onset of an eye movement served as an additional marker.

5 Conclusion

In order to robustly extract meaningful signals from noisy electrophysiological data, averaging over many similar trials is unavoidable. The nature of these data sets, i.e. correlations between electrodes, clustered time courses across trials and prior knowledge from the design of the experiment, suggests a number of more complex procedures

for cleaning data and enhancing the quality of the signal. We have discussed here the possibility to reduce the variability of the data by allowing for variable internal processing speeds. The specific application to EEG data does not limit the generality of the approach which may as well be used for other imaging techniques.

Acknowledgment. This work was supported by BMBF grant number 01GQ0432. Discussions with R. Schaback, M. Hasselhorn, J. Behrendt and H. Gibbons are gratefully acknowledged.

References

1. Flexer, A.: Data mining and electroencephalography. *Stat. Meth. Medical Res.* 9, 395 (2000)
2. Woody, C.D.: Characterization of an adaptive filter for the analysis of variable latency neuroelectric signals. *Medical and Biological Engineering and Computing* 5, 539–553 (1967)
3. Picton, T.W., Lins, O.G., Scherg, M.: The recording and analysis of event-related potentials. In: Boller, F., Grafman, J. (eds.) *Handbook of Neuropsychology*, pp. 3–73. Elsevier, Amsterdam (1995)
4. Gibbons, H., Stahl, J.: Response-time corrected averaging of event-related potentials. *Clinical Neurophysiology* 118, 197–208 (2007)
5. Schrobsdorff, H., Ihrke, M., Kabisch, B., Behrendt, J., Hasselhorn, M., Herrmann, J.M.: A Computational Approach to Negative Priming. *Connection Science* 19(3), 203–221 (2007)
6. Ihrke, M.: Negative priming and response-relation: Behavioral and electroencephalographic correlates. Master's thesis, U. Göttingen (2007), <http://www.psych.uni-goettingen.de/home/ihrke>
7. Haig, A.R., Gordon, E., Rogers, G., Anderson, J.: Classification of single-trial ERP subtypes. *Electroencephalography Clinical Neurophysiology* 94(4), 288–297 (1995)
8. Truccolo, W.A., Ding, M., Knuth, K.H., Nakamura, R., Bressler, S.L.: Trial-to-trial variability of cortical evoked responses. *Clinical Neurophysiology* 113(2), 206–226 (2002)
9. de Weerd, J.P.: A posteriori time-varying filtering of averaged evoked potentials. I. Introduction and conceptual basis. *Biological Cybernetics* 41(3), 211–222 (1981)
10. Nagelkerke, N.J.D., Strackee, J.: Some notes on the statistical properties of a posteriori Wiener filtering. *Biological Cybernetics* 33(2), 121–123 (1979)
11. Truccolo, W.A., Ding, M., Bressler, S.L.: Variability and interdependence of local field potentials. *Neurocomputing* 38(40), 983–992 (2001)
12. Ciganek, L.: Variability of the human visual evoked potential: normative data. *Electroencephalogr. Clin Neurophysiol.* 27(1), 35–42 (1969)
13. Basar, E., Gonder, A., Ozesmi, C., Ungan, P.: Dynamics of brain rhythmic and evoked potentials. *Biological Cybernetics* 20(3-4), 137–143 (1975)
14. Lange, D.H., Siegelmann, H.T., Pratt, H., Inbar, G.F.: Overcoming selective ensemble averaging: unsupervised identification of event-related brain potentials. *IEEE Transactions on Biomedical Engineering* 47(6), 822–826 (2000)
15. Tibshirani, R., Walther, G., Hastie, T.: Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B* 63(2), 411–423 (2001)
16. Woldorff, M.G.: Distortion of erp averages due to overlap from temporally adjacent erps: Analysis and correction. *Psychophysiology* 30, 98–119 (1993)
17. Myers, C., Rabiner, L.: A level building dynamic time warping algorithm for connected word recognition. *IEEE Transact. Acoustics, Speech, and Signal Proc.* 29, 284–297 (1981)
18. Keogh, E.J., Pazzani, M.J.: Derivative Dynamic Time Warping. In: *First SIAM International Conference on Data Mining (SDM 2001)* (2001)

Statistical Baselines from Random Matrix Theory

Marotesa Voultzidou¹ and J. Michael Herrmann²

¹ University of Crete, Department of Physics, P.O. Box 2208, Heraklion, Crete, Greece

² University of Edinburgh, Institute for Perception, Action and Behaviour
Informatics Forum, 11 Crichton Street, Edinburgh, EH8 9AB, U.K.

marotesa@physics.uoc.gr, michael.herrmann@ed.ac.uk

Abstract. Quantitative descriptors of intrinsic properties of imaging data can be obtained from the theory of random matrices (RMT). Based on theoretical results for standardized data, RMT offers a systematic approach to surrogate data which allows us to evaluate the significance of deviations from the random baseline. Considering exemplary fMRI data sets recorded at a visuo-motor task and rest, we show their distinguishability by RMT-based quantities and demonstrate that the degree of sparseness and of localization can be evaluated in a strict way, provided that the data are sufficiently well described by the pairwise cross-correlations.

1 Analysis of Image Sequences

In order to reveal significant features in empirical data there is often no other option than to compare certain data features to corresponding quantities in surrogate data [1]. By shuffling, boosting, randomizing or rearranging the data otherwise, a baseline is obtained that may reveal potentially interesting features in the data. The actual process of generation of surrogate data remains a matter of an on-going debate. For low randomization contrasts will be faint, while at strong shuffling any feature may appear significant. In high-dimensional problems the data are often sparse and thus not sufficiently representative for the underlying distribution. By reference to random matrices we suggest a more systematic framework for providing baselines to data features of potential interest and reduction of the data space such that other methods may become feasible for a further analysis of the data. We further present an approach to the identification of different experimental conditions by rating the difference of a quantity from Random Matrix Theory (RMT) calculated for the two conditions relatively to the respective deviations from the theoretical value. Of particular interest is, furthermore, that RMT provides descriptions of spatial properties of the data, that can be used for the discrimination of active and non-active brain voxels which forms an essential step in the analysis of fMRI data.

Data features of interest can thus be obtained by a comparison of the statistical properties of the eigenspectrum as well as the spatial properties of the eigenvectors of the data correlation matrix with those of random matrices. The data structures that do not conform to the RMT predictions deviate from universality and can be characterized as significant.

2 RMT for Data Processing

Random matrix theory studies ensembles of matrices that are given by a distribution over the set of all matrices. A frequently used ensemble is the Gaussian Orthogonal Ensemble (GOE) which stands for the set of matrices with entries drawn independently from a fixed Gaussian distribution. Interestingly, in the limit of high dimensions all matrices of the ensemble tend to share certain properties, e.g. the eigenvalue distribution. In this sense the properties of the ensemble are universal. Although RMT has been originally established in the field of physics [23] it has turned out to be applicable to a number of phenomena such as the stock market [4] and EEG recordings [5]. It permits the identification of features which are universally present in large classes of systems [6]. Theorems in RMT usually hold for matrices of infinite size. In many cases finite size corrections are available [7], which turn out to be helpful for the dimensionalities considered here. In addition, we include also numerical results of the finite size effects when comparing to the standard RMT ensembles.

The data from an fMRI experiment are given as an $(M \times T)$ -matrix \mathbf{X} where M denotes the number of voxels in an image (volume) and T is the length of the trial measured by the number of time steps. If X_i denotes the time series of activities of voxel i then its centered and variance-normalized version

$$D_{it} = \left(X_{it} - T^{-1} \sum_t X_{it} \right) \left(T^{-1} \sum_t \left(X_{it} - T^{-1} \sum_t X_{it} \right)^2 \right)^{-1/2} \quad (1)$$

gives rise to the data correlation matrix by the matrix product $\mathbf{C} = \mathbf{D}\mathbf{D}^\top$, where $\mathbf{D} = \{D_{it}\}$. Eigenvalues and the eigenvectors of \mathbf{C} are calculated by a singular value decomposition on the data matrix \mathbf{D} . The temporal average (1) may as well be replaced by a spatial average [8] to obtain spatial correlation matrices of the form $\mathbf{C}' = \mathbf{D}'^\top \mathbf{D}'$ with \mathbf{D}' being the spatially centered data. Correlation matrices are characterized by non-negativity and bounded entries. In order to satisfy this property an ensemble of random correlation matrices (RCE) is defined [9]. Matrices \mathbf{C} in RCE can be obtained from matrices \mathbf{B} by $\mathbf{C} = \mathbf{B}\mathbf{B}^\top$, where \mathbf{B} is a matrix of random elements with zero mean and unit variance. While the RCE is naturally better suited as a baseline to the data, the GOE will be used as a further reference.

After normalization, the bulk of the spectrum of the data correlation matrix conforms to the predictions of RMT up to finite-size corrections and can thus be understood as a stochastic invariant of the data. Singular data features that were found to deviate strongly from RMT can be interpreted as a consequence of physiological or task-related effects. The deviations serve as a criterion for the relevance of these directions in the data space. Alternatively, we analyzed what features contribute most to the deviations, similarly as in projection pursuit. In a sense, RMT yields a baseline against which the relevant, i.e. not the invariantly present data features, may become evident.

3 Higher-Order Data Characteristics from RMT

Universality of the results from RMT is revealed only after normalization. This is achieved by the so-called unfolding procedure [10] that generates a uniform eigenvalue distribution by subtracting the smooth part of the spectrum in order to reveal

fluctuations of the eigenvalues. In local unfolding the eigenvalues ϵ_i are transformed by $\epsilon_{i+1} = \epsilon_i + (\lambda_{i+1} - \lambda_i)/K_i$, where K_i is the local mean level spacing $K_i = \frac{1}{2k+1} \sum_{j=i-k}^{i+k} (\lambda_{j+1} - \lambda_j)$. The number $2k+1$ of consecutive level spaces in the running average is a free parameter of the model. Here the unfolded eigenvalues were obtained by performing local unfolding of the cumulative density function with $k = 20$.

A simple statistical quantity in RMT is the (nearest-neighbor) level-space distribution $P(s)$, i.e. the probability $P(s)$ that two adjacent eigenvalues are separated by a distance $s_i = \epsilon_{i+1} - \epsilon_i$. It provides information on short range spectral correlations. For the GOE, $P(s)$ obeys the Wigner surmise [11] $P(s) = (\pi/2)s \exp(-\pi s^2/4)$. Instead of calculating the level space distribution directly, the more robust integrated form

$$N(s) = \int P(s)ds \tag{2}$$

is used, resulting in $N_{RMT}(s) = 1 - \exp(-\pi s^2/4)$ for the GOE. Since $N_{RMT}(s)$ holds in the limit of infinite spectra, small differences are observed for finite-size ensembles.

Higher-order information can be found in the number variance $\Sigma^2(L)$ defined as the variance of the number of unfolded eigenvalues in an interval of length L .

$$\Sigma^2(L) = \langle N^2(L, \epsilon_0) \rangle - \langle N(L, \epsilon_0) \rangle^2 \tag{3}$$

$N(L, \epsilon_0)$ counts the number of levels in the interval $[\epsilon_0, \epsilon_0 + L]$ of unfolded eigenvalues and the averages are over ϵ_0 . For a GOE the number variance in the limit of large matrix size is given by $\Sigma^2(L) = 2\pi^2(\ln(2\pi L) + 1.5772 - \pi^2/8)$ [12,6]. Apart from finite size effects, Σ^2 saturates when L approaches the width of the unfolding procedure k . We will hence restrict comparisons involving $\Sigma^2(L)$ to L values below k .

4 Statistical Analysis of Eigenvectors

The concepts of the entropy localization length and the width of an eigenvector cover two main aspects, namely the sparsity of the eigenvectors and the relative location of their large components, respectively. The entropy localization length is based on the Shannon entropy [13,7,14] which is defined as

$$H_N^{(n)} = - \sum_{i=1}^N w_i^n \ln(w_i^n), \tag{4}$$

where $w_i^n \equiv (u_i^n)^2$ and u_i^n denotes the i^{th} component of a normalized eigenvector n ($n = 1, \dots, N$) of the data correlation matrix. The entropy $H_N^{(n)}$ gives a measure of the number of components in an eigenvector that are significantly large. In the case of extreme localization, i.e. $w_i^n = (0, \dots, 0, 1, 0, \dots, 0)$ we have $H_N^{(n)} = 0$. In the case of fully extended eigenvectors, i.e. $w_i^n = (1/N, \dots, 1/N)$, we find $H_N^{(n)} = \ln(N)$, which resembles a typical eigenvector of a random matrix. In the large- N limit Eq. 4 behaves as $\overline{H}_N = \ln(N\alpha/2) + 1/N$ where $\alpha \approx 0.96$ is a constant shift.

The entropy localization length l_H^n is defined as

$$l_H^n = N \exp\left(H_N^{(n)} - \overline{H}_N\right) = (2/\alpha) \exp\left(H_N^{(n)}\right). \tag{5}$$

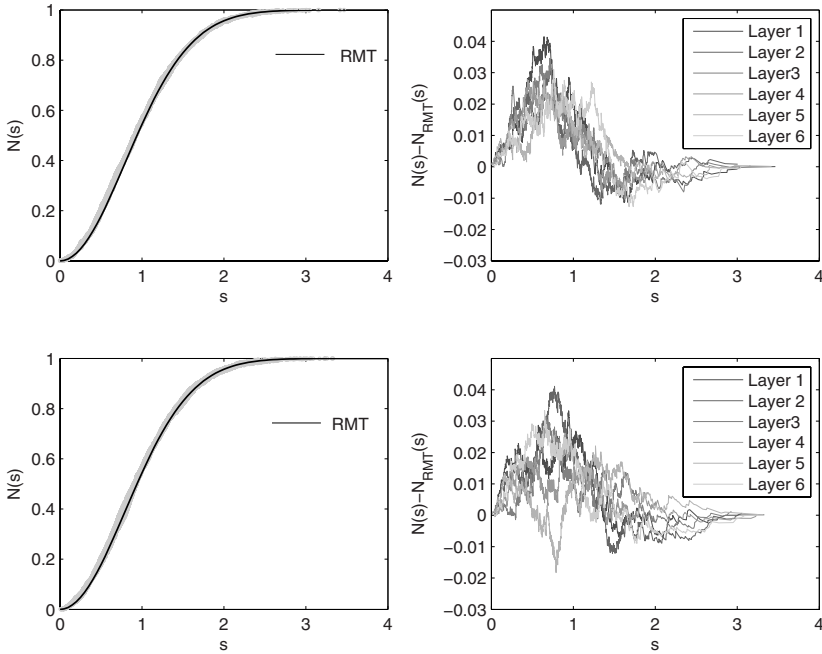


Fig. 1. Integrated level space distribution (left) and differences from the theoretical prediction (right) for the two fMRI data sets. (REST: top row, TASK: bottom row).

The entropy localization length (5) represents the effective number of large components of an eigenvector but without providing any information about their location. Eigenvectors of the same entropy localization length may have different structure depending on the location of the large/small components. Additional information can be expressed by the width of an eigenvector l_c (14)

$$l_c^{(n)} = \left\{ \sum_{i=1}^N w_i^n [(i_x - n_{cx}(n))^2 + (i_y - n_{cy}(n))^2] \right\}^{1/2}, \quad (6)$$

where (i_x, i_y) denotes the position of a voxel i and n_{cx}, n_{cy} are the components of the center of mass $n_c = \sum_i i w_i^n$. Small l_c implies localization of activity in a component.

5 Results for an Illustrative Data Set

Four sets of matrices were constructed corresponding to the ensembles RCE and GOE and the two conditions in the fMRI experiments. In the first condition the subject was in a resting state (REST) while in the second condition a visuo-motor task (TASK) was to be performed allowing for investigation on the significant features represented by physiological and stimulus influences respectively. The integrated level space distribution (2) does not indicate a significant deviation from the surrogate data, cf. Fig. 1.

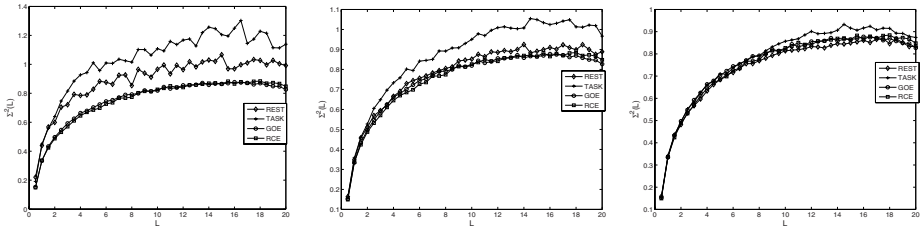


Fig. 2. (left) Number variance for the fMRI data sets (REST and TASK) and the random sets (GOE and RCE). Number variance with the $N_l = 10$ (middle) and the $N_l = 20$ (right) largest eigenvalues were discarded before the analysis.

This confirms earlier findings in EGG data [5] and suggests the consideration of more complex statistical quantities.

The fMRI data sets differ from the synthetic sets with respect to the number variance, while the difference between the synthetic data set is insignificant. The number variance of the TASK data deviates stronger from the GOE/RCE sets than the REST set which is mainly caused by the fact that stimulus-related activity is present in the eigenvectors of the TASK data. After discarding from the analysis a number $N_l = 10$ of the largest eigenvalues the REST results converge to the GOE/RCE behavior, see Fig. 2. The TASK results are affected only when the random “bulk” is exposed after the removal of at least $N_l = 20$ eigenvalues. By continuously increasing N_l the REST data set shows a sub-Gaussian behavior which implies that the bulk of the REST spectrum is more rigid than the GOE and RCE spectra. The rigid parts of the spectrum contribute little to $\Sigma^2(L)$. In the fMRI data sets the largest eigenvalues are least rigid. This is similar in the RCE, while in the GOE the spectrum loses rigidity at both edges. If we select only the rigid parts of the spectra the sub-Gaussian behavior of the REST data set is seen to vanish. Exclusion of an increased number of large eigenvalues from the analysis of the TASK data set shows a slow tendency to approach the GOE/RCE curve. This suggests that the rest of the spectrum still carries information which, however, cannot be separated by second-order statistics used in the current approach.

Fig. 3 shows the entropy localization length l_H and the width of the eigenvector l_c of the eigenvectors of the correlation matrix of one slice of the fMRI data and a GOE-like matrix for comparison. Both l_H and l_c of the RCE sample is similar to GOE and are not shown here. For the GOE sample, neither l_H nor l_c present a systematic dependency on the eigenvalue number. On the contrary, the entropy localization length l_H and the width of the eigenvector l_c of both fMRI data sets deviate from the random cases. These deviations are noticeable in the eigenvectors corresponding to the largest eigenvalues which exhibit small values of l_H and thus contain a small number of relative high components. The largest l_H value corresponds to the eigenvector with the largest eigenvalue and indicate global activation. Towards the center of the spectrum the eigenvectors show similar behavior as random vectors since the l_H values are around the number of their components.

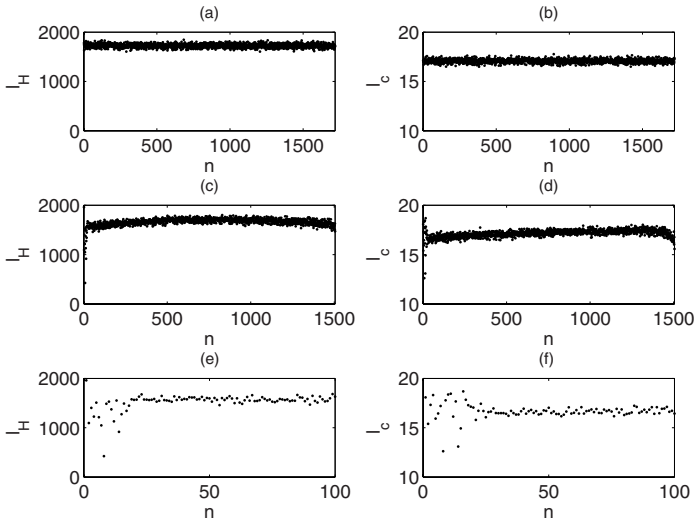


Fig. 3. Entropy localization length of the eigenvectors of a GOE-like matrix (a) and of the eigenvectors of a covariance matrix of fMRI data (c). In (a) the mean l_H (black line) is very close to the total number of eigenvectors N indicating that the eigenvectors are extended. In (b) and (d) the width of the eigenvectors of a GOE-like matrix and of those of a covariance matrix of fMRI data is shown respectively. In (e) and (f) magnification of plots (c) and (d) are shown respectively. The index n of the eigenvectors is in descending order according to the corresponding eigenvalue.

Additional information about the structure of the eigenvectors is provided by their width l_c . Eigenvectors with similar eigenvalues, in particular when these are large, exhibit different degrees of sparsity depending on the relative location of their high-value components. Although their entropy localization length is smaller than that of the bulk of the spectrum, their effective components are distributed from high localization to high extendedness. We propose to combine information given by the entropy localization length l_H and the width of the eigenvector l_c to select potentially interesting eigenvectors. In the first step the candidate vectors are these with a limited number of large components and are represented by small l_H . In the second step we check for localization and small l_c values. Because such features might be present in eigenvectors with small eigenvalues, the variance of the eigenvalues must be also taken into account. Fig. 4 illustrates this concept for the eigenvectors of the correlation matrix of one layer of the fMRI data and a GOE matrix for comparison.

We propose to combine information from entropy length l_H and eigenstate width l_c to select interesting eigenvectors. Firstly, candidate vectors are selected that comprised from a limited number of large components and have small l_H . Secondly, we check for localization and small l_c values. Because this combination might be present in eigenvectors with small eigenvalues, the variance of the eigenvalues must be also taken into account. Fig. 4 illustrates this concept for the eigenvectors of the correlation matrix of one layer of the fMRI data and a GOE matrix for comparison.

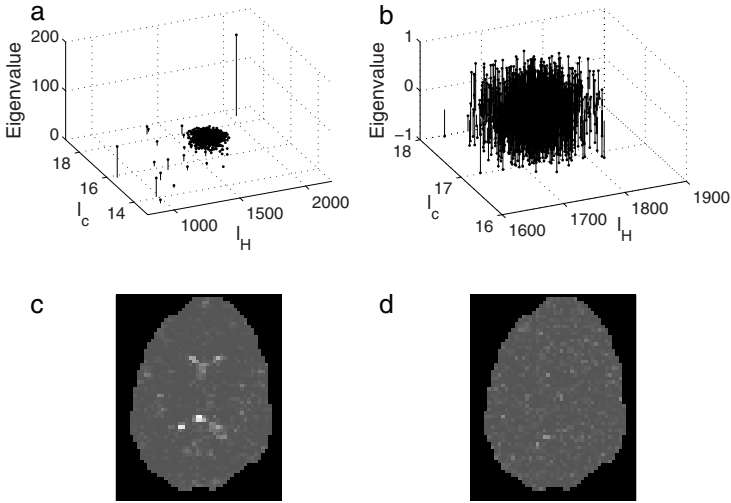


Fig. 4. Entropy localization length l_H as a function of the eigenvalue and the width of the eigenvectors l_c for a slice of fMRI data (a) and a GOE-like matrix (b). The height of the vertical lines represent the eigenvalues of the corresponding eigenvectors. In bottom row a localized eigenvector with small l_H , small l_c and large eigenvalue is shown (c) and an extended eigenvector with relatively high l_H and l_c from the bulk of the spectrum (d).

6 Discussion

We have presented an approach based on random matrix theory that provides a criterion to discriminate between noise-like and relevant components in fMRI data. The criterion is based on the extensively studied statistical properties of the eigenvalues and the eigenvectors of random matrices [6,12,13,14] as compared with those of data correlation matrices. The data components that share the same behavior with random vectors can be interpreted as unstructured noise while those that deviate from the RMT predictions are considered as potentially interested for further processing. The approach was also evaluated to simulated data sets of several conditions and various localization properties and the results were again compared with those from the GOE/RCE ensembles. The statistical analysis of the eigenvalues and the eigenvectors of the artificial sets clearly revealed the correct number of relevant eigenvectors. Although the simulated data sets were rather simplistic they provide good evidence of the applicability of the current approach in correlational analysis of fMRI data.

The analysis of the statistical properties of the eigenvalues of the data correlation matrices showed that the level space distribution for the two experimental conditions does not exhibit any significant deviations from RMT predictions. On the other hand, the number variance gives evidence of the significance of the data features. The number variance of both data sets deviates from RMT but it is more prominent in the condition when the subjects were engaged in a sensory-motor activity. The exclusion of subsets of eigenvalues shows that these deviations occurred predominantly at large eigenvalues,

which became visible by the differences of the Σ^2 between the two fMRI data sets as well as between the fMRI data sets and the GOE/RCE ensembles.

We also studied the spatial properties of the eigenvectors in terms of the number of their effective components and their localization given by l_H and l_c respectively. The combination of l_H and l_c offers an additional criterion for selecting the appropriate number of significant eigenvectors since stimulus related ones are expected to be of a limited number of large components gathered in localized regions.

In the present study we have been working with rather abstract features. This has the advantage that these can be compared across different trials, sessions and subjects serving thus the data meta-analysis. Further, it should be noticed that while the implementation of the RMT approach was on spatial PCA it can be applied as well to spatial PCA or to matrices obtained by ICA.

Acknowledgments. The authors wish to thank the NMR GmbH Göttingen and J. B. Poline for the data, and S. Dodel, T. Geisel, G. Tsironis and T. Kottos for inspiring discussions. This work was supported by EU and the Greek Ministry of Education (E/ITEAEK II) and by the Marie Curie program. J.M.H. is PI of the BCCN Göttingen and is associated to the MPI for Dynamics and Self-Organization, Bunsenstr. 10, 37073 Göttingen.

References

1. Theiler, J., Eubank, S., Longtin, A., Galdrikian, B., Farmer, J.D.: Testing for nonlinearity in time series: The method of surrogate data. *Physica D* 58 (1992)
2. Wigner, E.P.: Random matrix theory in physics. *SIAM Rev.* 9, 1–23 (1967)
3. Mehta, M.L.: *Random Matrices*. Academic Press, Boston (1991)
4. Plerou, V., Gopikrishnan, P., Rosenow, B., Amaral, L.A.N., Guhr, T., Stanley, H.E.: Random matrix approach to cross correlations in financial data. *Phys. Rev. E* 65 (2002)
5. Šeba, P.: Random matrix analysis of human EEG data. *Phys. Rev. Lett.* 91(19) (2003)
6. Brody, T.A., Flores, J., French, J.B., Mello, P.A., Pandey, A., Wong, S.S.M.: Random-matrix physics: spectrum and strength fluctuations. *Rev. Mod. Phys.* 53(3) (1981)
7. Casati, G., Guarneri, I., Izrailev, F., Scharf, R.: Scaling behavior of localization in quantum chaos. *Phys. Rev. Lett.* 64(1) (1990)
8. Dodel, S., Herrmann, J.M., Geisel, T.: Comparison of temporal and spatial ica in fmri data analysis. In: *Proc. ICA 2000, Helsinki, Finland*, pp. 543–547 (2000)
9. Voultsidou, M., Dodel, S., Herrmann, J.M.: Feature evaluation in fmri data using random matrix theory. *Comput. Visual. Sci.* 10(2), 99–105 (2007)
10. Manfredi, V.R.: Level density fluctuations of interacting bosons. *Nuovo Cimento Lettere* 40, 135 (1984)
11. Wigner, E.P.: On the distribution of the roots of certain symmetric matrices. *Ann. of Math.* 67, 325–328 (1958)
12. Guhr, T., Müller-Groelling, A., Weidenmüller, H.A.: Random-matrix physics: Spectrum and strength fluctuations. *Phys. Rep.* 299(190) (1998)
13. Izrailev, F.M.: Intermediate statistics of the quasi-energy spectrum and quantum localization of classical chaos. *J. Phys. A: Math. Gen.* 22, 865–878 (1989)
14. Luna-Acosta, G.A., Méndez-Bermúdez, J.A., Izrailev, F.M.: Periodic chaotic billiards: Quantum-classical correspondence in energy space. *Phys. Rev. E* 64 (2001)

Adaptive Classification by Hybrid EKF with Truncated Filtering: Brain Computer Interfacing

Ji Won Yoon¹, Stephen J. Roberts¹, Matthew Dyson², and John Q. Gan²

¹Pattern Analysis and Machine Learning Group
Department of Engineering Science, University of Oxford, UK
{jwoon, sjrob}@robots.ox.ac.uk

<http://www.robots.ox.ac.uk/~parg>

²Department of Computing and Electronic Systems
University of Essex, UK

{mdyson, jqgan}@essex.ac.uk

<http://dces.essex.ac.uk/staff/jqgan/>

Abstract. This paper proposes a robust algorithm for adaptive modelling of EEG signal classification using a modified Extended Kalman Filter (EKF). This modified EKF combines Radial Basis functions (RBF) and Autoregressive (AR) modeling and obtains better classification performance by truncating the filtering distribution when new observations are very informative.

Keywords: Extended Kalman Filter, Informative Observation, Logistic Classification, Truncated Filtering.

1 Introduction

The goal of Brain Computer Interfacing (BCI) is to enable people with severe neurological disabilities to operate computers by manipulation of the brain's electrical activity rather than by physical means. It is known that the generation and control of electrical brain activity (the electroencephalogram or EEG) signals for a BCI system often requires extensive subject training before a reliable communication channel can be formed between a human subject and a computer interface [1,2]. In order to reduce the overheads of training and, importantly, to cope with new subjects, adaptive approaches to the core data modelling have been developed for BCI systems. Such adaptive approaches differ from the typical methodology, in which an algorithm is trained *off-line* on retrospective data, in so much that the process of 'learning' is continuously taking place rather than being confined to a section of 'training data'. In the signal processing and machine learning communities this is referred to as *sequential classification*. Previous research in this area applied to BCI data has used *state space modelling* of the time series [3,4,5]. In particular Extended Kalman Filter (EKF) approaches have been effective [3,5]. In these previous models, based upon the EKF, the variances of the observation noise and hidden state noise are re-estimated using

a maximum-likelihood style framework, ultimately due to the work of Jazwinski [7]. In previous work [6], we developed a simple, computationally practical, modification of the conventional EKF approach in which we marginalise (integrate out) the unknown variance parameters for variances by applying Bayesian conjugate priors [8]. This enabled us to avoid the need for spurious parameter estimation steps which decrease the robustness of the algorithm in highly non-stationary and noisy regions of data. The EKF offers one framework for approximating the non-linear linkage between observations and decisions (via the logistic function). In all this prior work, however, the output function assumed the target labels (the decision classes) to be time independent, i.e. there is no explicit Markov dependency from one decision to the next. This is a poor model for the characteristics of the BCI interface, at the sample rate we operate at (over 0.2s intervals), in which sequences of labels of the same class persist for several seconds before making a transition.

The algorithm presented here employs both a set of non-linear basis functions (thus making it a dynamic Radial Basis Function (RBF) classifier) and an autoregressive (AR) model to tackle this Markov dependency of the labeling. In addition, our approach modifies a filtering step by truncating the filtering distribution in terms of new observations. This modified filtering step provides a robust algorithm when given labels (i.e. as training data) are very informative. Moreover, this algorithm does not have to approximate the convolution of a logistic function and Normal distribution using a probit function, which can lead to poor performance for the EKF [5,6].

2 Mathematical Model

We consider an observed input stream (i.e. a time series) \mathbf{x}_t at time $t = 1, 2, \dots, T$ which we project into a non-linear basis space, $\mathbf{x}_t \rightarrow \varphi(\mathbf{x}_t)$. We also (partially) observe $z_t \in \{0, 1\}$ such that $Pr(z_t = 1 | \mathbf{x}_t, \mathbf{w}_t) = g(f(\mathbf{x}_t, \mathbf{w}_t))$ where $g(\cdot)$ can be logistic model or probit link function which takes the *latent* variables to the outcome decision variable. In this paper, we use the logistic function, i.e. $g(s) = 1/(1 + e^{-s})$. The state space model for the BCI system can hence be regarded as a hierarchical model in that the noise of observations influences the model indirectly through the logistic regression model $g(\cdot)$. Such indirect influence makes for a more complicated model, but we can circumvent much of this complexity by forming a three stage state space model and by introducing an auxiliary variable y_t . The latter variable acts so as to link the indirect relationships between observations and the logistic regression model given by

$$\begin{aligned} p(z_t | y_t) &= g(y_t)^{z_t} (1 - g(y_t))^{1-z_t} \\ y_t &= a \mathbf{B} \mathbf{w}_t^T \varphi_t(\mathbf{x}_t) + (1 - a) y_{t-1} + v_t \\ \mathbf{w}_t &= \mathbf{A} \mathbf{w}_{t-1} + \mathbf{L} \mathbf{h}_t \end{aligned} \tag{1}$$

where $v_t \sim \mathcal{N}(0, \kappa)$ and $\mathbf{h}_t \sim \mathcal{N}(\mathbf{0}, \tau \mathbf{I})$. The variable a denotes a convex mixing ratio between the Radial Basis Function (RBF) and the Autoregressive (AR) model. To simplify notation, we use φ_t instead of $\varphi_t(\mathbf{x}_t)$ i.e.

$$\varphi_t = \varphi_t(\mathbf{x}_t) = \left[\mathbf{x}_t^T, \{\phi_t^{(1)}(\mathbf{x}_t)\}^T, \dots, \{\phi_t^{(N_b)}(\mathbf{x}_t)\}^T, 1 \right]^T \tag{2}$$

and $\phi_t^{(i)}(\mathbf{x}_t)$ is the response of the i th Gaussian basis function for $i \in \{1, \dots, N_b\}$ at time t . Here, N_b is the number of Gaussian basis functions and the basis functions are chosen in random. In adaptive classification, there are two steps in our state space model which perform model inference:

- **Prediction:** $p(\mathbf{w}_t, y_t | \mathbf{z}_{1:t-1}, \mathbf{x}_{1:t-1})$
- **Filtering:** $p(\mathbf{w}_t, y_t | \mathbf{z}_{1:t}, \mathbf{x}_{1:t})$

after which we obtain $\bar{y}_{t|t} = \int_{y_t} y_t \left[\int_{\mathbf{w}_t} p(\mathbf{w}_t, y_t | \mathbf{z}_{1:t}, \mathbf{x}_{1:t}) d\mathbf{w}_t \right] dy_t$.

2.1 Prediction

$$\begin{aligned} & p(\mathbf{w}_t, y_t | \mathbf{z}_{1:t-1}, \mathbf{x}_{1:t-1}) \\ &= \int p(\mathbf{w}_t, y_t, \mathbf{w}_{t-1}, y_{t-1} | \mathbf{z}_{1:t-1}, \mathbf{x}_{1:t-1}) d\mathbf{w}_{t-1} dy_{t-1} \\ &= \int p(y_t | \mathbf{w}_t, y_{t-1}) p(\mathbf{w}_t | \mathbf{w}_{t-1}) p(\mathbf{w}_{t-1}, y_{t-1} | \mathbf{z}_{1:t-1}, \mathbf{x}_{1:t-1}) d\mathbf{w}_{t-1} dy_{t-1} \\ &= \int \left[\int_{\kappa} p(y_t | \mathbf{w}_t, y_{t-1}, \kappa) p(\kappa) d\kappa \right] \left[\int_{\tau} p(\mathbf{w}_t | \mathbf{w}_{t-1}, \tau) p(\tau) d\tau \right] \\ &\quad \times p(\mathbf{w}_{t-1}, y_{t-1} | \mathbf{z}_{1:t-1}, \mathbf{x}_{1:t-1}) d\mathbf{w}_{t-1} dy_{t-1} \tag{3} \\ &\approx \int \mathcal{N} \left(y_t; a\varphi_t^T \mathbf{B} \mathbf{w}_t + (1-a)y_{t-1}, \left(\frac{\alpha_{\kappa}^*}{\beta_{\kappa}} \right)^{-1} \right) \mathcal{N} \left(\mathbf{w}_t; \mathbf{A} \mathbf{w}_{t-1}, \left(\frac{\alpha_{\tau}^*}{\beta_{\tau}} \right)^{-1} \mathbf{L} \mathbf{L}^T \right) \\ &\quad \times \mathcal{N} \left(\begin{bmatrix} y_{t-1} \\ \mathbf{w}_{t-1} \end{bmatrix}; \mu_{t-1|t-1}, \Sigma_{t-1|t-1} \right) d\mathbf{w}_{t-1} dy_{t-1} \\ &= \int \mathcal{N} \left(\begin{bmatrix} y_t \\ \mathbf{w}_t \end{bmatrix}; \begin{bmatrix} 1-a & a\varphi_t^T \mathbf{B} \mathbf{A} \\ \mathbf{0} & \mathbf{A} \end{bmatrix} \begin{bmatrix} y_{t-1} \\ \mathbf{w}_{t-1} \end{bmatrix}, \Sigma \right) \mathcal{N} \left(\begin{bmatrix} y_{t-1} \\ \mathbf{w}_{t-1} \end{bmatrix}; \mu_{t-1|t-1}, \Sigma_{t-1|t-1} \right) d\mathbf{w}_{t-1} dy_{t-1} \\ &= \mathcal{N} \left(\begin{bmatrix} y_t \\ \mathbf{w}_t \end{bmatrix}; \mathbf{S} \mu_{t-1|t-1}, \Sigma + \mathbf{S} \Sigma_{t-1|t-1} \mathbf{S}^T \right) \tag{4} \end{aligned}$$

where

$$\begin{aligned} \mathbf{S} &= \begin{bmatrix} 1-a & a\varphi_t^T \mathbf{B} \mathbf{A} \\ \mathbf{0} & \mathbf{A} \end{bmatrix}, \Sigma = \begin{bmatrix} \Sigma_y & \Sigma_c \\ \Sigma_c & \Sigma_w \end{bmatrix}, \\ \Sigma_y &= \mathbf{E}[(y_t - \mathbf{E}(y_t))(y_t - \mathbf{E}(y_t))^T] = a^2 \varphi_t^T \mathbf{B} \mathbf{L} \left(\frac{\alpha_{\tau}^*}{\beta_{\tau}} \right)^{-1} \mathbf{L}^T \mathbf{B}^T \varphi_t + \left(\frac{\alpha_{\kappa}^*}{\beta_{\kappa}} \right)^{-1}, \\ \Sigma_w &= \mathbf{E}[(\mathbf{w}_t - \mathbf{E}(\mathbf{w}_t))(\mathbf{w}_t - \mathbf{E}(\mathbf{w}_t))^T] = \mathbf{L} \left(\frac{\alpha_{\tau}^*}{\beta_{\tau}} \right)^{-1} \mathbf{L}^T, \\ \Sigma_c &= \mathbf{E}[(y_t - \mathbf{E}(y_t))(\mathbf{w}_t - \mathbf{E}(\mathbf{w}_t))^T] = a\varphi_t^T \mathbf{B} \mathbf{L} \left(\frac{\alpha_{\tau}^*}{\beta_{\tau}} \right)^{-1} \mathbf{L}^T. \tag{3} \end{aligned}$$

2.2 Filtering

In the filtering step, our EKF uses a new observation z_t . In general, $p(\mathbf{w}_t, y_t | \mathbf{z}_{1:t}, \mathbf{x}_{1:t})$ is used with the following form:

$$\begin{aligned}
 p(\mathbf{w}_t, y_t | \mathbf{z}_{1:t}, \mathbf{x}_{1:t}) &\propto p(z_t, \mathbf{w}_t, y_t | \mathbf{z}_{1:t-1}, \mathbf{x}_{1:t-1}) \\
 &= p(z_t | y_t) p(\mathbf{w}_t, y_t | \mathbf{z}_{1:t-1}, \mathbf{x}_{1:t-1}).
 \end{aligned}
 \tag{4}$$

It is difficult to obtain a full distribution for the EKF filtering step so previous approaches approximated the convolution between the normal distribution and a logistic function using the probit function [5][6]. However, if the new observation is very informative, filtering can be improved without the need for the approximation, which is one of the major sources of non-linearity in logistic EKF filtering. We can obtain better filtering distributions by truncation as we do not need the convolution approximation. Hence we obtain

$$\begin{aligned}
 &p(\mathbf{w}_t, y_t | \mathbf{z}_{1:t}, \mathbf{x}_{1:t}) \\
 &= p(\mathbf{w}_t, y_t | \mathbf{z}_{1:t-1}, \mathbf{x}_{1:t}, z_t) = \begin{cases} p(\mathbf{w}_t, y_t | \mathbf{z}_{1:t-1}, \mathbf{x}_{1:t-1}) \mathbf{I}_{y_t \geq 0}(y_t) & \text{if } z_t = 1 \\ p(\mathbf{w}_t, y_t | \mathbf{z}_{1:t-1}, \mathbf{x}_{1:t-1}) \mathbf{I}_{y_t < 0}(y_t) & \text{if } z_t = 0 \end{cases} \\
 &\quad \text{where } \mathbf{I}_{\mathbf{C}}(y_t) = \begin{cases} 1, & \text{if } y_t \text{ satisfies } \mathbf{C} \\ 0, & \text{otherwise,} \end{cases} \\
 &= \begin{cases} p(\mathbf{w}_t | y_t, \mathbf{z}_{1:t-1}, \mathbf{x}_{1:t-1}) p(y_t | \mathbf{z}_{1:t-1}, \mathbf{x}_{1:t-1}) \mathbf{I}_{y_t \geq 0}(y_t) & \text{if } z_t = 1 \\ p(\mathbf{w}_t | y_t, \mathbf{z}_{1:t-1}, \mathbf{x}_{1:t-1}) p(y_t | \mathbf{z}_{1:t-1}, \mathbf{x}_{1:t-1}) \mathbf{I}_{y_t < 0}(y_t) & \text{if } z_t = 0 \end{cases} \\
 &\approx \begin{cases} p(\mathbf{w}_t | y_t, \mathbf{z}_{1:t-1}, \mathbf{x}_{1:t-1}) [p(y_t | \mathbf{z}_{1:t-1}, \mathbf{x}_{1:t-1}) \mathbf{I}_{y_t \geq 0}(y_t)] & \text{if } z_t = 1 \\ p(\mathbf{w}_t | y_t, \mathbf{z}_{1:t-1}, \mathbf{x}_{1:t-1}) [p(y_t | \mathbf{z}_{1:t-1}, \mathbf{x}_{1:t-1}) \mathbf{I}_{y_t < 0}(y_t)] & \text{if } z_t = 0 \end{cases} \\
 &= \begin{cases} \mathcal{N}(\mathbf{w}_t; \hat{\mu}_{w,t|t-1}, \hat{\Sigma}_{w,t|t-1}) [\mathcal{N}(y_t; \mu_{y,t|t-1}, \Sigma_{y,t|t-1}) \mathbf{I}_{y_t \geq 0}(y_t)] & \text{if } z_t = 1 \\ \mathcal{N}(\mathbf{w}_t; \hat{\mu}_{w,t|t-1}, \hat{\Sigma}_{w,t|t-1}) [\mathcal{N}(y_t; \mu_{y,t|t-1}, \Sigma_{y,t|t-1}) \mathbf{I}_{y_t < 0}(y_t)] & \text{if } z_t = 0 \end{cases} \\
 &\quad \text{where } \begin{cases} \hat{\mu}_{w,t|t-1} = \mu_{w,t|t-1} + \Sigma_{c,t|t-1} \Sigma_{y,t|t-1}^{-1} (y_t - \mu_{y,t|t-1}) \\ \hat{\Sigma}_{w,t|t-1} = \Sigma_{w,t|t-1} - \Sigma_{c,t|t-1}^T \Sigma_{y,t|t-1}^{-1} \Sigma_{c,t|t-1} \end{cases} \\
 &\approx \mathcal{N}(\mathbf{w}_t; \hat{\mu}_{w,t|t-1}, \hat{\Sigma}_{w,t|t-1}) \mathcal{N}(y_t; \mu_{y,z_t}, \Sigma_{y,z_t}) \text{ for } z_t \in \{0, 1\}
 \end{aligned}$$

where

$$\begin{aligned}
 \mu_{y,z_t=1} &= \mathbf{E}_{\mathcal{N}(y_t; \mu_{y,t|t-1}, \Sigma_{y,t|t-1}) \mathbf{I}_{y_t \geq 0}(y_t)}(y_t) \\
 \mu_{y,z_t=0} &= \mathbf{E}_{\mathcal{N}(y_t; \mu_{y,t|t-1}, \Sigma_{y,t|t-1}) \mathbf{I}_{y_t < 0}(y_t)}(y_t) \\
 \Sigma_{y,z_t=1} &= \mathbf{V}_{\mathcal{N}(y_t; \mu_{y,t|t-1}, \Sigma_{y,t|t-1}) \mathbf{I}_{y_t \geq 0}(y_t)}(y_t) \\
 \Sigma_{y,z_t=0} &= \mathbf{V}_{\mathcal{N}(y_t; \mu_{y,t|t-1}, \Sigma_{y,t|t-1}) \mathbf{I}_{y_t < 0}(y_t)}(y_t)
 \end{aligned}
 \tag{5}$$

and $\mathbf{E}(\cdot)$ and $\mathbf{V}(\cdot)$ denote mean and covariance of the truncated normal distribution respectively. The calculation of the mean and covariance of the truncated normal distribution is described in the appendix. Therefore, we have

$$p(\mathbf{w}_t, y_t | \mathbf{z}_{1:t}, \mathbf{x}_{1:t}) = \mathcal{N} \left(\begin{bmatrix} y_t \\ \mathbf{w}_t \end{bmatrix}; \mu_{t|t}, \Sigma_{t|t} \right) = \mathcal{N} \left(\begin{bmatrix} y_t \\ \mathbf{w}_t \end{bmatrix}; \begin{bmatrix} \mu_{y,t|t} \\ \mu_{w,t|t} \end{bmatrix}, \begin{bmatrix} \Sigma_{y,t|t} & \Sigma_{c,t|t} \\ \Sigma_{c,t|t} & \Sigma_{w,t|t} \end{bmatrix} \right)
 \tag{6}$$

where

$$\begin{aligned}
 \mu_{y,t|t} &= \mu_{y_t, z_t} \\
 \Sigma_{y,t|t} &= \Sigma_{y_t, z_t} \\
 \Sigma_{c,t|t} &= \Sigma_{c,t|t-1} \Sigma_{y,t|t-1}^{-1} \Sigma_{y_t, z_t} \\
 \mu_{w,t|t} &= \mu_{w,t|t-1} + \Sigma_{c,t|t-1} \Sigma_{y,t|t-1}^{-1} (\mu_{y, z_t} - \mu_{y,t|t-1}) \\
 \Sigma_{w,t|t} &= \Sigma_{w,t|t-1} + \Sigma_{c,t|t-1} \Sigma_{y,t|t-1}^{-1} \Sigma_{y, z_t} \Sigma_{y,t|t-1}^{-1} \Sigma_{c,t|t-1} - \Sigma_{c,t|t-1} \Sigma_{y,t|t-1}^{-1} \Sigma_{c,t|t-1}^T
 \end{aligned} \tag{7}$$

3 Results for Experimental Data Set

3.1 Data Acquisition

Data used in this experiment consisted of two channels of EEG, recorded at 256Hz placed over the central portion of the head and one channel of muscle electrical activity (EMG), recorded at 1024Hz over the muscles of the right fore-arm. The EMG was then down-sampled to 256Hz and muscle contraction strength for movement and non-movement detection was evaluated via a simple windowed peak and trough detection; this then formed a movement / non-movement label.

3.2 Feature Extraction and Basis Formation

The second reflection coefficient of a second-order autoregressive (AR) model [10] were calculated over each EEG signal once every 78ms using a sliding one-second-

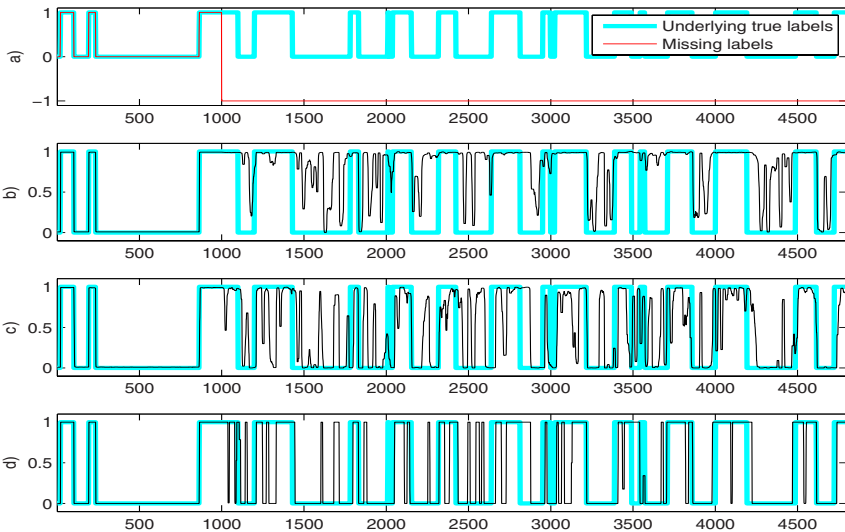


Fig. 1. Comparison of different EKF algorithms: a) trajectories with true labels (blue) and missing labels (red), b) Standard EKF (fixed $\tilde{\tau}$ and $\tilde{\kappa}$ are used), c) Modified EKF (marginalising both τ and κ), d) hybrid EKF using marginalisation and truncation

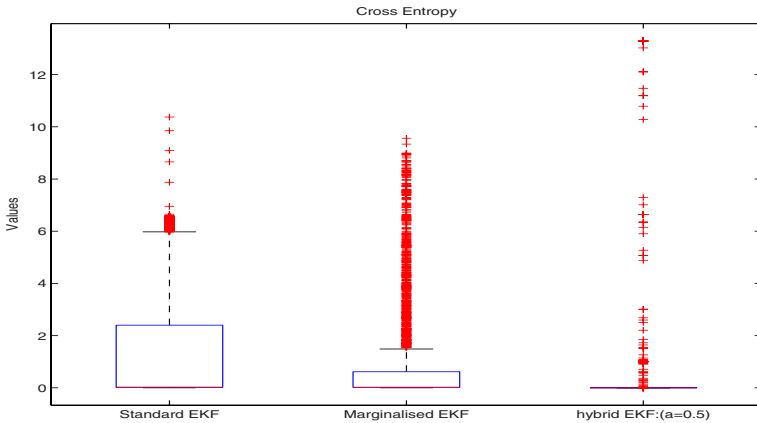


Fig. 2. Comparison of Cross Entropy: the method proposed in this paper outperforms other EKF approaches

long window, forming a set of feature vectors \mathbf{x}_t . These vectors were projected into a non-linear latent space using a set of Gaussian basis functions ($N_b = 10$), providing the stream of φ_t .

3.3 Performance Comparison

In this paper, hyper-parameters defining prior distributions over τ and κ are given by $\alpha_\tau = 2, \beta_\tau = 100, \alpha_\kappa = 5,$ and $\beta_\kappa = 0.1$. Since there is no proper model in dynamics for our model, we used a transition kernel based on random walks so that the matrices of Eq. (11) are known as $\mathbf{A} = \mathbf{B} = \mathbf{L} = \mathbf{I}_{N_w}$ where \mathbf{I}_{N_w} denotes an $N_w \times N_w$ size identity matrix and N_w is the number of elements of \mathbf{w}_t . For comparison, fixed $\tilde{\tau}$ and $\tilde{\kappa}$ are chosen by $\tilde{\tau} = \mathbf{E}(\tau|\alpha_\tau, \beta_\tau) = \int \tau p(\tau|\alpha_\tau, \beta_\tau) d\tau$ and $\tilde{\kappa} = \mathbf{E}(\kappa|\alpha_\kappa, \beta_\kappa) = \int \kappa p(\kappa|\alpha_\kappa, \beta_\kappa) d\kappa$ where $\tilde{\tau}$. Also, $p(\mathbf{w}_0) = \mathcal{N}(\cdot; \mu_0, \Sigma_0)$ where $\mu_0 = \mathbf{0}$ and $\Sigma_0 = \tau_0 \mathbf{I}$ respectively. Here $\tau_0 = 2000000$. Fig. 1 shows the comparison of several methods. The first plot of the figure shows the labels: inactive ($z_t = 0$), active ($z_t = 1$) and unknown labels ($z_t = -1$) in the movement respectively. The algorithm receives no label information after $t = 1000$ samples.

Fig. 2 demonstrates the comparison of the cross entropy of the several methods. The cross entropy of the t th sample is defined by

$$\mathbf{E}_t^{cross} = -\log p(z_t|\tilde{z}_t) = -\log \{ \tilde{z}_t^{z_t} (1 - \tilde{z}_t)^{1-z_t} \} \tag{8}$$

where z_t and \tilde{z}_t represent for underlying labels and the posterior of estimated labels respectively.

4 Conclusion and Future Work

In this paper, we propose a dynamic Bayesian model for adaptive classification in time series based on a Bayesian version of the modified Extended Kalman filter

which combines Radial basis function and autoregressive model. The AR model explains the Markov dependency of the output function so that it provides the possibility of improvement in conventional EKF classification. This proposed algorithm also employs truncation of filtering distribution to obtain a better classifier when the observation is very informative. The critical data-specific sensitivity of our model is to the hyper-parameter a . In principle this can be inferred from a section of labelled data or set via knowledge of the typical within-state times. This requires a more extensive testing against a larger number of data sets than presented in this pilot study and this will be the focus of future work.

Acknowledgment

This project is supported by UK EPSRC funding to whom we are most grateful.

References

1. Wopaw, J.R., McFarland, D.J., Neat, D.J., Forneris, C.A.: An EEG-based brain-computer interface for cursor control. *Electroencephalogr. Clin. Neurophysiol.* 78, 252–259 (1991)
2. Pfurtscheller, G., Flotzinger, D., Kalcher, J.: Brain-computer interface: a new communication device for handicapped persons. *Journal of Microcomputer Applications archive* 16(3), 293–299 (1993)
3. Sykacek, P., Roberts, S.J., Stokes, M.: Adaptive BCI Based on Variational Bayesian Kalman Filtering: An Empirical Evaluation. *IEEE Transactions on Biomedical Engineering* 51(5), 719–727 (2004)
4. Penny, W.D., Roberts, S.J., Curran, E.A., Stokes, M.: EEG-based communication: a pattern recognition approach. *IEEE Transactions on Rehabilitation Engineering* 8(2), 214–216 (2000)
5. Lowne, D.R., Roberts, S.J., Garnett, R.: Sequential Non-stationary Dynamic Classification. *Machine Learning* (submitted, 2008)
6. Yoon, J., Roberts, S.J., Dyson, M., Gan, J.Q.: Sequential Bayesian Estimation for Adaptive Classification. *Multisensor Fusion and Integration for Intelligent Systems* (accepted for publication, 2008)
7. Jazwinski, A.H., Bailie, A.E.: Adaptive filtering Interim report. *NASA Technical Reports* (March 1, 1967)
8. Bernardo, J.M., Smith, A.F.M.: *Bayesian Theory*. Wiley, Chichester (1994)
9. Bishop, C.M.: *Pattern Recognition and Machine Learning*, 1st edn. Springer, Heidelberg (1988)
10. Pardey, J., Roberts, S.J., Tarassenko, L.: A Review of Parametric Modelling Techniques for EEG Analysis. *Med. Eng. Phys.* 18(1), 2–11 (1996)

Appendix

4.1 Laplace Approximation

In Eq. (3), the non-Gaussian distribution generated from the marginalisation and logistic model are approximated by Gaussian distributions using the Laplace

approximation. Let $\pi(\mathbf{w}_t)$ be the non-Gaussian distribution of interest. Then, we have

$$\pi(\mathbf{w}_t) = (2\pi)^{N_{\mathbf{w}}/2} \frac{\beta_\tau^{\alpha_\tau}}{\Gamma(\alpha_\tau)} \frac{\Gamma(\alpha_\tau^*)}{\left[\beta_\tau + \frac{1}{2}(\mathbf{w}_t - \mathbf{w}_{t-1})^T(\mathbf{w}_t - \mathbf{w}_{t-1})\right]^{\alpha_\tau^*}}. \tag{9}$$

In order to find the mode of $\pi(\mathbf{w}_t)$, we use log of the distribution given by

$$\mathcal{L}_{\mathbf{w}_t} = \log \pi(\mathbf{w}_t) = -\alpha_q^* \log \left[\beta_\tau + \frac{1}{2}(\mathbf{w}_t - \mathbf{w}_{t-1})^T(\mathbf{w}_t - \mathbf{w}_{t-1}) \right] + c \tag{10}$$

where $c = -N_w/2 \log(2\pi) + \alpha_\tau \log \beta_\tau - \log \Gamma(\alpha_\tau) + \log(\alpha_\tau^*)$. Using the first derivate of $\mathcal{L}_{\mathbf{w}_t}$ in terms of \mathbf{w}_t , we have

$$\frac{dL}{d\mathbf{w}_t} = -\frac{\alpha_\tau^*}{\beta_\tau + \frac{1}{2}(\mathbf{w}_t - \mathbf{w}_{t-1})^T(\mathbf{w}_t - \mathbf{w}_{t-1})}(\mathbf{w}_t - \mathbf{w}_{t-1}) = 0 \tag{11}$$

and a mode can be easily obtained by $\mathbf{w}_t = \mathbf{w}_{t-1}$. In addition, using the second derivative of the $\mathcal{L}_{\mathbf{w}_t}$, we can obtain the covariance of the approximated distribution as follows:

$$\left. \frac{d^2 L}{d\mathbf{w}_t^2} \right|_{\mathbf{w}_t = \mathbf{w}_{t-1}} = \frac{\alpha_\tau^*}{\beta_\tau}. \tag{12}$$

Now, we have an approximated distribution

$$\pi(\mathbf{w}_t) \approx N \left(\mathbf{w}_t; \mathbf{w}_{t-1}, \left(\frac{\alpha_\tau^*}{\beta_\tau} \right)^{-1} \right). \tag{13}$$

4.2 Truncated Normal Distribution

Suppose we condition on $\mathbf{Y}_t \in \theta = [\theta_1, \theta_2]$, where $-\infty < \theta_1 < \theta_2 < \infty$. When we have the normal distribution with mean $\mu = \mu_{y_t|t-1}$ and standard deviation $\sigma = \sqrt{\Sigma_{y_t|t-1}}$, the conditional density of \mathbf{Y}_t is

$$p(y_t|\theta) = \frac{\frac{1}{\sigma} \phi\left(\frac{y_t - \mu}{\sigma}\right)}{\Phi\left(\frac{\theta_2 - \mu}{\sigma}\right) - \Phi\left(\frac{\theta_1 - \mu}{\sigma}\right)}, \theta_1 \leq y_t \leq \theta_2 \tag{14}$$

where $\phi(\cdot)$ and $\Phi(\cdot)$ denote normal distribution function and normal cumulative distribution function respectively.

$$\begin{aligned} \mu_{y_t, z_t=1} &= \mathbf{E}[\mathbf{Y}_t | \mathbf{Y}_t >= 0] = \mu + \sigma \frac{\phi\left(-\frac{\mu}{\sigma}\right)}{1 - \Phi\left(-\frac{\mu}{\sigma}\right)} \\ \mu_{y_t, z_t=0} &= \mathbf{E}[\mathbf{Y}_t | \mathbf{Y}_t < 0] = \mu - \sigma \frac{\phi\left(-\frac{\mu}{\sigma}\right)}{\Phi\left(-\frac{\mu}{\sigma}\right)} \\ \Sigma_{y_t, z_t=1} &= \mathbf{V}[\mathbf{Y}_t | \mathbf{Y}_t >= 0] = \sigma^2 \left\{ 1 + \frac{-\frac{\mu}{\sigma} \phi\left(-\frac{\mu}{\sigma}\right)}{1 - \Phi\left(-\frac{\mu}{\sigma}\right)} - \left[\frac{\phi\left(-\frac{\mu}{\sigma}\right)}{1 - \Phi\left(-\frac{\mu}{\sigma}\right)} \right]^2 \right\} \\ \Sigma_{y_t, z_t=0} &= \mathbf{V}[\mathbf{Y}_t | \mathbf{Y}_t < 0] = \sigma^2 \left\{ 1 + \frac{\frac{\mu}{\sigma} \phi\left(-\frac{\mu}{\sigma}\right)}{\Phi\left(-\frac{\mu}{\sigma}\right)} - \left[\frac{\phi\left(-\frac{\mu}{\sigma}\right)}{\Phi\left(-\frac{\mu}{\sigma}\right)} \right]^2 \right\}. \end{aligned} \tag{15}$$

Improving the Relational Evaluation of XML Queries by Means of Path Summaries

Sherif Sakr

National ICT Australia (NICTA), Sydney, Australia
sherif.sakr@nicta.com.au

Abstract. XML query languages such as XQuery, XSLT and SQL/XML are mainly dependent on XPath as the search and extraction language. XPath expressions often define complicated navigations which require expensive query processing costs especially when they are executed over large collections of XML documents. In this paper, we describe an approach of exploiting materialized XPath views to improve the efficiency of relational query processing of XML queries. The main contribution of this paper is to show that an intuitive and *very cheap* Data Guide synopsis of XML path summaries in addition a *light-weight* tracing of XPath steps can significantly reduce the XML query-evaluation costs in the relational hosts. Our experiments shows that the overhead introduced by the use of path summaries and an additional path identifier of node-based relational encoding of the XML documents is negligible but can result in significant reduction of the processing costs of relational evaluation of XML queries.

1 Introduction

XML has been acknowledged as the defacto standard for data representation and exchange over the World Wide Web. In recent years, XML has found practical application in numerous domains including data interchange, streaming data and data storage. As XML continues to grow in popularity, large repositories of XML documents are going to emerge, and users are likely to pose increasingly more complex queries on these data sets. As a simple XML query language but with sufficient expressive power, XPath has become increasingly popular and is also used as the sub-language of other XML query languages such as XSLT, XQuery and SQL/XML [4]. XPath expressions often define complicated navigation over the XML trees. Hence, the evaluation of XPath expressions may require expensive query processing costs especially when they are executed over large collections of documents. Several techniques have been proposed to speed up the evaluation of XPath queries. Indexing techniques [12,13], structural join algorithms [3,8,16] and materialized views [1,2,17] are three well-known approaches that can significantly accelerate the performance of the evaluation of XML queries. In this paper we consider the problem of improving the efficiency of evaluating XML queries on relational databases using materialized views in a special form of very efficient Data Guide-based XML path summaries. The work of this paper is based on three main building blocks:

- 1) A purely relational approach of evaluating XML Queries [7,9,10,11].
- 2) Data Guide-based summary synopsis for XML documents [5].
- 3) A variant of the projection paths technique presented by Marian and Siméon in [14] to statically analyse and infer the relevant paths used within a given XQuery expression during compilation time.

We propose a framework for exploiting these three components to expedite the processing of XML queries. Aligned with the compositional nature of the XQuery expressions where sub-expressions are combined with each other to form the final query, the XQuery-to-SQL translation approach described in [7,9,10,11] has the same compositional nature. We describe a simple and intuitive approach to *speed-up* the SQL evaluation of XPath sub-expressions in the XML queries using a new introduced pre-computed or materialized *Guide Node* notion. This guide node represents the connection between each group of nodes in the source XML document and their associated representative *PathID* in the Data Guide-based path summary synopsis. We believe that our approach can be applied to any other relational-based implementation of XML queries in a similar way.

The remainder of this paper is organized as follows. Section 2 presents a brief overview of the relevant aspects of the basis relational approach of XQuery processing. Section 3 describes the used Data Guide-based path summaries synopsis and introduces the *Guide Node* notion. Section 4 presents the enhanced SQL translation of XPath sub-expression in the XQuery context using the inferred *Guide Node* information. Experimental evaluation for the *Guide Node*-based SQL translation of XQuery expressions is presented in Section 5 before we conclude in Section 6.

2 Relational Evaluation of XML Queries

The XQuery-to-SQL Translation approach presented in [7,9,10,11] is based on three main components:

- 1) An efficient *pre/size/level* relational encoding of XML documents [12].
- 2) The *loop-lifting* compilation technique for translating the XQuery expressions into relational algebraic plans [7,10,11].
- 3) A translation mechanism for translating the intermediate algebraic plan into SQL translation script [10].

In the following subsections we give a brief overview of the three main components. For more details we refer the reader to the following literature [7,9,10,11]

2.1 Relational Encoding of XML Documents

Having an appropriate XML storage scheme is a crucial part for any relational implementation of an XQuery processor. The work of [9,10] has been based on the efficient and scalable *pre/size/level* relational encoding scheme described by Grust in [12]. In this encoding the information of the XML node hierarchy is mapped to a relational table which preserves the structural relationship between the XML nodes. The encoding is obtained by a single sequential document read

Table 1. XPath axes evaluation conditions

XPath Axis	Axis Conditions
self	$pre(x) = pre(y) \wedge kind(y) \neq att$
attribute	$parent(y) = pre(x) \wedge kind(y) = att$
parent	$parent(x) = pre(y) \wedge kind(y) \neq att$
child	$parent(y) = pre(x) \wedge kind(y) \neq att$
descendant	$pre(y) > pre(x) \wedge pre(y) \leq pre(x) + size(x) \wedge kind(y) \neq att$
descendant-or-self	$pre(y) \geq pre(x) \wedge pre(y) \leq pre(x) + size(x) \wedge kind(y) \neq att$
ancestor	$pre(y) < pre(x) \wedge pre(x) \leq pre(y) + size(y) \wedge kind(y) \neq att$
ancestor-or-self	$pre(y) \leq pre(x) \wedge pre(x) \leq pre(y) + size(y) \wedge kind(y) \neq att$
following	$pre(y) > pre(x) + size(x) \wedge kind(y) \neq att$
following-sibling	$pre(y) > pre(x) \wedge parent(y) = parent(x) \wedge kind(y) \neq att$
preceding	$pre(y) + size(y) < pre(x) \wedge kind(y) \neq att$
preceding-sibling	$pre(y) < pre(x) \wedge parent(y) = parent(x) \wedge kind(y) \neq att$

where a pre-order rank is assigned for each node v in the XML tree and a node-tuple descriptor is inserted in the encoding relation which contains the following information: $Size(v)$ represents the number of nodes in the sub-tree below the node v , $Level(v)$ represents the number of intermediate levels between the root node and a node v , $Parent(v)$ stores the pre-order rank for each node's parent, $Kind(v)$ stores the kind of the encoded document node (document, element, attribute, text, name space, or processing instruction node), $Name(v)$ stores the tag name for the element nodes, and $Value(v)$ stores the atomic values for nodes with the kind of *text* or *attribute* and stores *null* for the nodes of the other types.

Based on the defined encoding scheme, the evaluation conditions of the 12 XPath axes could be defined as depicted in Table 1. A sample interpretation of the XPath axes evaluation conditions is for example: given two XML nodes x and y in an XML tree T , y is a *descendant* of x if and only if $(pre(y) > pre(x)) \wedge (pre(y) \leq pre(x) + size(x)) \wedge (kind(y) \neq att)$. Using these evaluation conditions, translating the XPath expressions into SQL Queries is a straightforward process. An XPath expression with a series of location steps represented as $S_1/S_2/.../S_n$ is converted into a series of n join queries between n instances of the *pre/size/level* encoding relation where the node sequence output by axis step S_i is the context node sequence for the subsequent step S_{i+1} .

2.2 Algebraic Compilation of XQuery Expressions

The *loop lifting* compilation technique [10,11] compiles XQuery expressions into equivalent relational query plans using a very simple form of *tuple-based relational algebra* that can efficiently fit within the capabilities of SQL-based systems [10,11]. The principal idea behind the compilation scheme is that every XQuery expression occurs in the scope of an iteration. The iterations of each scope are encoded by a column *iter* in the associated relational representation. A loop of n iterations is represented by a relation **loop** with a single column *iter* of n values (1,2,...,n) and the compilation of variables bound (x) in the iterations of FLWOR expression is represented by encoding all bindings of x in a single relation where each tuple of the encoding relation (i, p, x) indicates that for the i -th iteration, the item at position p stores the value x .

2.3 SQL Translation of Relational XQuery Plans

The translation of relational algebraic plans into its equivalent SQL scripts is easily achieved by traversing DAG-shaped intermediate relational XQuery plans in a bottom up fashion and then translating each algebraic operator into its equivalent SQL evaluation step using a set of well-defined SQL translation templates for the algebraic operators. The generated SQL scripts are standard SQL:1999 code which can be executed on any conventional RDBMS [9].

3 Path-Based Materialization Views

3.1 XML Paths Summary

In our approach we use an XML path summary synopsis which represents a concrete implementation for the *Data Guide* summary structure [5]. The root node of the data guide represents the root element of the document and every node is a *Guide node*. Every *Guide node* represents a *correspondent* node for *all* nodes in the source XML document which are sharing the same rooted path starting from the root node. Figure 1 illustrates an example of the *data guide* tree synopsis of the well-known "auction.xml" XML document of the XMark benchmark [15]. We construct the data guide path summary representation of the XML document during the normal parsing and shredding process of the XML document into the encoding *pre/size/level* relational scheme. Data Guide is known to be very efficient in terms of memory usage. The size of Data Guide is proportional to the number of *distinct* root-to-leaf paths and is not related to the XML document size. Hence, the Data Guide size of an XMark document instance with a size of 10 MB is equivalent to the Data Guide size of another instance with a size of 100 MB.

As illustrated in Figure 1, each *Guide Node* is identified with its pre-order rank according to its position in the summary tree structure. Hence, each *Guide Node* with its associated pre-order rank represents a form of *PathID* for their correspondent nodes in the source XML documents. We extended the *pre/size/level* encoding relation of the source XML document with an additional attribute to store the *PathID* information for each node. Additionally, we build a *partitioned B-tree* index [6] over the (*PathID*, *pre*) attributes to form the basis of a materialized view which establishes the link between each node in the source XML document and its correspondent *Guide Node* (*PathID*) in the Data Guide. The main limitation of the Data Guide is that it does not capture the order information of the XML document and does not support the tracking of order-sensitive XPath axes. However, the needs of using these axes are in practice much less than the order-insensitive axes they are also out of the scope of this paper.

3.2 Relational Analysis of Projection Paths (Guide Nodes)

In the algebra of the *loop-lifting* compilation, the XPath evaluator operator (\lrcorner) is responsible for representing the XPath steps. It receives a relation for the

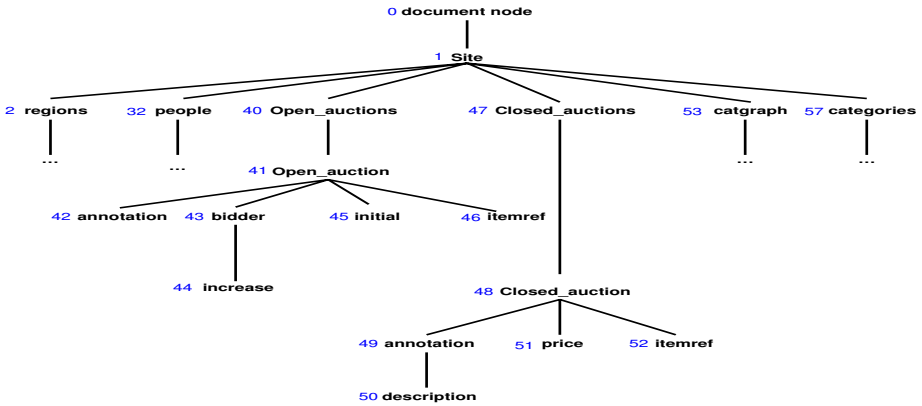


Fig. 1. Data Guide of XMark document

associated live nodes of the XML fragment and an input *context* relation (*ctx*) with the schema (*iter, item*) where the *iter* column represents the iteration scope of each node and the *item* column stores the *node identifiers* of the input context nodes. In our approach we statically analyze the XPath evaluator operator during compile-time to infer the relevant *PathID(s)* and to keep track of the *guide node(s)* of *node identifiers* through the sequence of the context relations. The idea of guide node(s) tracing is very similar to the idea presented by Marian and Siméon in [14]. However, in [14] they use the information of the relevant paths to minimize the main memory requirement of the Galax XQuery processor while in our approach it is used to minimize the cost of evaluating a sequence of XPath steps. In our approach, we use an annotation for the *item* columns to store the *node identifiers* of the context nodes which we name as a *Guide Node* annotation. An *item* column of the context relation (*ctx*) annotated with the *Guide Node* property *X* indicates that the list of the *node identifiers* stored in the *item* are corresponding to the node with the pre-order rank *X* of the *Data Guide*. Having an input context relation where the *item* column is annotated by the *Guide Node* property *X*. Applying the XPath evaluator operator ($\sigma_{item:(\alpha,n)}$) over the context relation *ctx* annotates the *item* column of the resulting context relation with the *Guide Node* property equal to *Y*, where *Y* is the set of the pre-order rank(s) of the resulting node(s) from applying the path step (α, n) where α represents the XPath step and *n* represents the node test over the *Guide Node* *X* in the XML *Data Guide*. More detailed explanation for this annotation process will be presented in the example of Section 4.

4 Rewriting SQL Translation of XML Queries Using Guide Node Annotation

In Section 2, we described the conventional approach for translating the XPath expression into SQL queries using the *pre/size/level* encoding. In this section we

represent our mechanism of rewriting the SQL evaluation of XPath expressions using materialized *PathID* information and the inferred *Guide Node* annotation in order to show that the rewritten query can be more efficient if it utilizes the knowledge of the structural summary. Our mechanism is based on the observation that we can rewrite the SQL translation of the rooted XPath expressions in the relational algebraic plan using the *Guide Node* information of the last path step in the rooted sequence. To illustrate let us consider the following example:

```

S1
for $x in doc("auction.xml")/site/open_auctions/open_auction
return $x
    
```

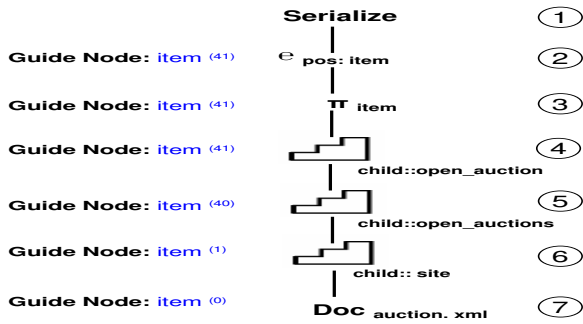


Fig. 2. Relational algebraic plan of the XQuery expression S1

Figure 2 illustrates the relational algebraic plan of the XQuery expressions S1. Based on the *pre/size/level* encoding, the conventional SQL translation of the sequence of XPath steps of the operators (7),(6),(5),(4) is:

```

1  SELECT d4.pre as item
2  FROM document AS d1,document AS d2,document AS d3,document AS d4
3  WHERE d1.kind = 'Doc' and d1.name='auction.xml'
4  AND d2. pre >= d1.pre AND d2.pre <=d1.pre + d1.size
5  AND d2.level=d1.level+1 AND d2.name='site' AND d2.kind='Elem'
6  AND d3. pre >= d2.pre AND d3.pre <=d2.pre + d2.size
7  AND d3.level=d2.level+1 AND d3.name='open_auctions' AND d3.kind='Elem'
8  AND d4. pre >= d3.pre AND d4.pre <=d3.pre + d3.size
9  AND d4.level=d3.level+1 AND d4.name='open_auction' AND d4.kind='Elem';
    
```

where line number 3 represents the evaluation conditions of the operator (7), lines 4 and 5 represent the evaluation conditions of the operator (6), lines 6 and 7 represent the evaluation conditions of the operator (5), lines 8 and 9 represent the evaluation conditions of the operator (4).

The inference process of the guide node information starts with the algebraic operator which represents the call of the XQuery built-in function $fn : doc$ (7). At this time, the *item* column of the operator (7) is annotated with the root guide node (0) of the data guide. The inference processes of the guide node information is achieved by traversing the relational algebraic plan in a bottom-up fashion where each occurrence of an XPath navigation step $\lceil Item:(\alpha,n)$ is applied

over the Data Guide. For example, using the Data Guide of XMark document (Figure 1) and by applying the path step $\mathcal{E}_{Item:(child,site)}$, the algebraic operator ⑥ is annotated with the guide node (1). Similarly, the algebraic operators ⑤, ④ are annotated with the guide node (40,41) respectively. Hence, using the inferred *Guide Node* information, we can discard the operators ⑦, ⑥, ⑤ and use the *Guide Node* property of the algebraic operator ④ to rewrite the SQL translation of the same combined XPath steps translation pattern as follows:

```
SELECT pre AS item FROM document WHERE path_Id = 41;
```

Clearly, the rewriting mechanism with the *Guide Node* information can achieve a significant improvement in the execution times of the evaluation of XPath steps especially in the cases of rewriting long sequence of XPath steps as well as in the cases of processing large XML documents. In addition, rewriting the SQL evaluation of XPath expression using the *guide node* information could be considered as a form of *schema aware optimization*. By applying the *guide node* inference mechanism, we could avoid the evaluation of XPath expressions which yield to an empty sequence of context nodes. Applying the path steps of such path expression over the *Data Guide* tree will yield to an algebraic operator with an empty set of *guide node* annotations. Such instances of the algebraic operators could be simply pruned and translated into a very cheap *SELECT* statement from an *empty* table instead of using the relatively expensive conventional SQL translation. Moreover, although the above information may also be inferred from a DTD or XML Schema definition for the structure of the XML document (*if it is available*), the *Data Guide* information is still more precise as it only accounts for the paths occurring in the data.

5 Experiments

In this section we report on the experimental evaluation our approach. The experiments are performed on a DB2 9 server installation on a PC with 3.2 GHZ Intel Pentium 4 processor and 1 GB of main memory storage. We used an XMark document instance with a size of 30 MB (*scaling factor* 0.25). Figure 3 indicates the percentage of *speed-up* improvement on the execution times of the SQL-based relational evaluation of the 20 XMark benchmark queries [15] using the inferred *Guide Node* information over the conventional SQL translation. The reported percentage of speed up improvements are computed using the formula: $(1 - \frac{G}{C})\%$ where G represents the execution time of the SQL translation with the *Guide Node* information and C represents the execution time of the conventional translation. The results of this experiment confirm the efficiency of the *Guide Node* based translation over the conventional SQL translation. The execution times of most of the queries could significantly benefit from the guide node information. Q_{15} and Q_{16} gain the highest percentage of improvement because of the the long sequences of path steps. The lowest percentage of performance are gained by the queries Q_8 to Q_{12} because the dominating bulks of execution times are consumed by the join operations.

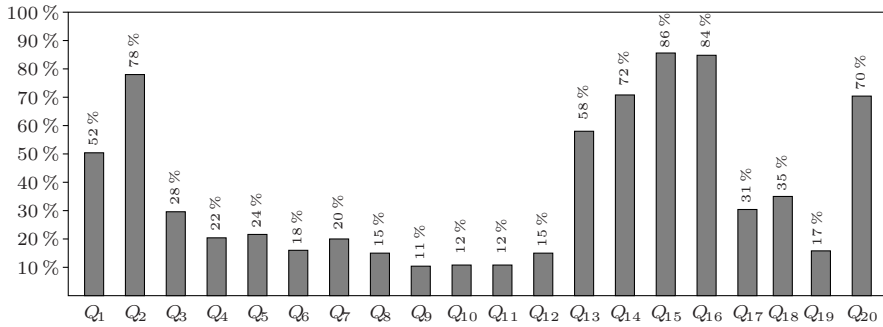


Fig. 3. The *speedup improvement* for evaluating XMark queries set using path summaries

6 Conclusion

We have described an approach for accelerating the evaluation of XPath expression in the context of XML queries using a special form of materialized view represented in the form of partitioned B-tree index. The proposed approach is very cheap in terms of main memory requirement and additional overhead of the storage space. It uses a light-weight analysis of the relational XQuery plans to directly access the nodes of the relevant paths. Our experiments have shown that this approach can significantly improve the execution time of XPath expressions and consequently the execution time of any relational evaluation of any container query language such as: XSLT, XQuery or SQL/XML.

References

- Balmin, A., Özcan, F., Beyer, K., Cochrane, R., Pirahesh, H.: A Framework for Using Materialized XPath Views in XML Query Processing. In: VLDB (2004)
- Barta, A., Consens, M., Mendelzon, A.: XML Query Optimization Using Path Indexes.. In: XIME-P (2004)
- Bruno, N., Koudas, N., Srivastava, D.: Holistic twig joins: optimal XML pattern matching. In: SIGMOD (2002)
- Eisenberg, A., Melton, J.: Advancements in SQL/XML. SIGMOD Record 33(3) (2004)
- Goldman, R., Widom, J.: DataGuides: Enabling Query Formulation and Optimization in Semistructured Databases.. In: VLDB (1997)
- Graefe, G.: Sorting And Indexing With Partitioned B-Trees.. In: CIDR (2003)
- Grust, T.: Purely Relational FLWORs. In: XIME-P (2005)
- Grust, T., Keulen, M., Teubner, J.: Staircase Join: Teach a Relational DBMS to Watch its (Axis) Steps.. In: VLDB (2003)
- Grust, T., Mayr, M., Rittinger, J., Sakr, S., Teubner, J.: A SQL:1999 Code Generator for the Pathfinder XQuery Compiler. In: SIGMOD (2007)
- Grust, T., Sakr, S., Teubner, J.: XQuery on SQL Hosts. In: VLDB (2004)
- Grust, T., Teubner, J.: Relational Algebra: Mother Tongue – XQuery: Fluent. In: Twente Data Management Workshop (TDM) (2004)

12. Grust, T.: Accelerating XPath location steps.. In: SIGMOD (2002)
13. Jiang, H., Lu, H., Wang, W., Xu Yu, J.: XParent: An Efficient RDBMS-Based XML Database System. In: ICDE (2002)
14. Marian, A., Siméon, J.: Projecting XML Documents.. In: VLDB (2003)
15. Schmidt, A., Waas, F., Kersten, M., Carey, M., Manolescu, I., Busse, R.: XMark: A Benchmark for XML Data Management. In: VLDB (2002)
16. Wu, Y., Patel, J., Jagadish, H.V.: Structural Join Order Selection for XML Query Optimization.. In: ICDE (2003)
17. Xu, W., Meral, Z.: Rewriting XPath queries using materialized views. In: VLDB (2005)

Identification of the Inverse Dynamics Model: A Multiple Relevance Vector Machines Approach

Chuan Li^{1,2}, Xianming Zhang¹, Shilong Wang², Yutao Dong¹, and Jing Chen¹

¹ Engineering Research Center for Waste Oil Recovery of Ministry of Education, Chongqing Technology and Business University, 400067 Chongqing, China

² College of Mechanical Engineering, Chongqing University, 400044 Chongqing, China
chuanli@21cn.com, xmzhang@ctbu.edu.cn, slwang@cqu.edu.cn,
mse20021c@sohu.com

Abstract. Relevance vector machines (RVM) is a machine learning approach with good nonlinear approximation capacity and generalization performance. In order to solve the inverse model for nonlinear systems, a multiple relevance vector machines (MRVM) based inverse dynamics model identification approach was presented. The input and output variables were allocated into multiple calculational subspaces according to their differential orders for the system. The RVM was put forward to identify the influence of the outputs to the inputs with a certain differential order in each subspace. Moreover, another RVM was delivered to connect all subspaces, such that the MRVM based inverse dynamics identification model for the nonlinear systems was constructed. At last it was applied to identify the inverse dynamics of a high temperature exchanger for the generator. And the result validates the effectiveness of the proposed approach.

Keywords: Inverse dynamics, Relevance vector machines, Nonlinear system, Sparse Bayesian learning, Identification.

1 Introduction

Inverse system theory is an effective tool for general nonlinear system control through finding the *Inverse* of the system [1]. And the identification of the inverse dynamics model is vital for its applications, such as adaptive inverse control, direct inverse control, α -th order inverse control and so on [2, 3].

Briefly, the identification of the inverse dynamics model is to calculate the input variables according to the output processes of the system. Due to its complexity, the nonlinear system modeling is very hard, which results in more difficult identification for the inverse dynamics. So it has been one of the bottlenecks for the research on the inverse system theory. Recently, some methods have been developed to identify the inverse dynamics model, such as artificial neural networks (ANN) and support vector machines (SVM) [4, 5].

According to the inverse system theory, the inputs of a system can be inverted through analyzing the historic inputs and outputs sequence. To model this inverting process, an inverse dynamics model of the system can be obtained. For an arbitrary nonlinear Multiple Input and Multiple Output (MIMO) discrete system G , there are [6]:

$$G(\mathbf{y}^{(i)}, \mathbf{y}^{(i-1)}, \dots, \mathbf{y}, \mathbf{u}^{(j)}, \mathbf{u}^{(j-1)}, \dots, \mathbf{u}) = 0 \quad (1)$$

where $\mathbf{y} = [y_1, y_2, \dots, y_p]^T$, is the output vector, $\mathbf{u} = [u_1, u_2, \dots, u_q]^T$, is the input vector, $i = -r, -r+1, \dots, -1, 0, 1, \dots, m, j = 1, 2, \dots, n$ and $\mathbf{y}^{(i)}$ is the i -th order differential of \mathbf{y} .

Suppose that the nonlinear discrete system G is reversible. When inverting the inputs from a certain output processes of a system, the input vector \mathbf{u} can be computed from:

$$\mathbf{u} = f(\mathbf{y}^{(i)}, \mathbf{y}^{(i-1)}, \dots, \mathbf{y}, \mathbf{u}^{(j)}, \mathbf{u}^{(j-1)}, \mathbf{u}^{(1)}) \quad (2)$$

Sometimes the integral of the outputs should be taken into account. However, it can only influence the physical structure and do not affect the identification method for the inverse dynamics. In traditional identification methods, the differential variables of the system were taken to the inputs of the mapping networks directly, which transformed the multi-order dynamic time model to be the static spatial model. Unfortunately, as $\mathbf{u} = [u_1, u_2, \dots, u_q]^T$ is a q -dimensional vector and $\mathbf{y} = [y_1, y_2, \dots, y_p]^T$ p -dimensional, the input dimension of a multi-order differential mapping networks is $p \times i + q \times j$. When the number of the input and output variables is bigger, the nonlinear mapping methods, either ANN or SVM, involve high-dimensional matrix calculation, which lead to the complicated calculations, morbid matrix, even modeling failure.

Instead of employing one single model, the approach that connects multiple sub-models so as to improve the regression performance has been used for modeling researches [7]. To reduce the dimension while identifying the inverse dynamics, the complicated model was converted to be the synthesis of multiple simple sub-models. The inputs and outputs were allocated into multiple calculational subspaces according to their differential orders for the system. And the relevance vector machines (RVM) was employed to identify the influence of output processes to the inputs with a certain differential order in each subspace. Moreover, another RVM was delivered to connect all subspaces, such that the multiple relevance vector machines (MRVM) based inverse dynamics identification model was constructed.

2 Modeling Approach Based on MRVM

2.1 Decompositions of the Identification Model

Two crucial factors should be taken into consideration for the identification of the inverse dynamics: the positive dynamics model (1) and the expression of the inputs from the positive dynamics model (2). But it is very difficult to obtain such a closed form for an actual system. RVM is a machine learning approach firstly proposed by M.T. Tipping [8]. As is researched, the SVM does suffer from a number of disadvantages, notably the absence of probabilistic outputs, the requirement to estimate a trade-off parameter and the need to utilize Mercer kernel functions. So the RVM, a Bayesian treatment of a generalized linear model of identical functional form to the SVM is offered to map the relation of (2) instead of a closed form. It suffers from none of the above disadvantages, and examples demonstrate that for comparable generalization performance, the RVM requires dramatically fewer kernel functions.

According to differential orders of variables, the inputs of (2) are segmented so as to establish $i+1$ independent calculational subspaces. In each subspace, the k -th order ($k \in [0, i]$) outputs of the system and the inputs ($\mathbf{y}^{(k)}, \mathbf{u}^{(j)}, \mathbf{u}^{(j-1)}, \dots, \mathbf{u}^{(1)}$) are selected as the inputs of the sub-model. And a RVM is offered to estimated \mathbf{u} from this subspace.

2.2 The Nonlinear Identification/Regression Algorithm of the RVM

Let nonlinear functions $f_0(\cdot), f_1(\cdot), f_2(\cdot), \dots, f_i(\cdot)$ be the regression of $(\mathbf{y}^{(k)}, \mathbf{u}^{(k)})$ with different orders 0, 1, ..., i to \mathbf{u} . Owing to the nonlinear generalization performance of RVM, $i+1$ independent RVM networks are proposed to map functions $f_0(\cdot), f_1(\cdot), f_2(\cdot), \dots, f_i(\cdot)$:

$$\begin{cases} RVM_1 : \hat{\mathbf{u}} = f_0(\mathbf{y}, \mathbf{u}^{(j)}, \mathbf{u}^{(j-1)}, \mathbf{u}^{(1)}) + \boldsymbol{\varepsilon}_0 \\ RVM_2 : \hat{\mathbf{u}} = f_1(\mathbf{y}^{(1)}, \mathbf{u}^{(j)}, \mathbf{u}^{(j-1)}, \mathbf{u}^{(1)}) + \boldsymbol{\varepsilon}_1 \\ \dots \\ RVM_{i+1} : \hat{\mathbf{u}} = f_i(\mathbf{y}^{(i)}, \mathbf{u}^{(j)}, \mathbf{u}^{(j-1)}, \mathbf{u}^{(1)}) + \boldsymbol{\varepsilon}_i \end{cases} \tag{4}$$

In the sparse Bayesian learning we are given a set of examples of input vectors $\mathbf{x} = \{x_1, x_2, \dots, x_n, \dots, x_N\}^T$ along with corresponding output $\mathbf{t} = \{t_1, t_2, \dots, t_n, \dots, t_N\}^T$. The learning target of RVM is to learn a model of the regression on the inputs with the objective of making accurate predictions of t for values of x . Suppose that the predictions comprises an unknown functions and some noises [9]:

$$\mathbf{t} = y(\mathbf{x}; \mathbf{w}) + \boldsymbol{\varepsilon} \tag{5}$$

Where, $\mathbf{w} = \{w_1, w_2, \dots, w_i, \dots, w_M\}^T$ is the weights of the model. $\boldsymbol{\varepsilon} = \{ \varepsilon_1, \varepsilon_2, \dots, \varepsilon_n, \dots, \varepsilon_N \}^T$ is the noise. $y(\mathbf{x}; \mathbf{w})$ is the function that is given from:

$$y(\mathbf{x}; \mathbf{w}) = \sum_{i=1}^M w_i \Phi(\mathbf{x}) + w_0 \tag{6}$$

where $\Phi(\mathbf{x})$ is the ‘‘design’’ matrix, i.e.

$$\Phi(\mathbf{x}) = \{ \phi_1(\mathbf{x}), \phi_2(\mathbf{x}), \dots, \phi_i(\mathbf{x}), \dots, \phi_M(\mathbf{x}) \}^T \tag{7}$$

In the SVM $\Phi(\mathbf{x})$ must be Mercer kernel functions. However, RVM has not this limitation. The spare Bayesian framework assumes an independent zero-mean Gaussian noise model, with variance σ^2 , giving a multivariate Gaussian likelihood of the target vector \mathbf{t} :

$$p(\mathbf{t} | \mathbf{w}, \sigma^2) = (2\pi\sigma^2)^{-N/2} \exp \left\{ -\frac{\|\mathbf{t} - \mathbf{w}\Phi\|^2}{2\sigma^2} \right\}^T \tag{8}$$

The prior over the parameters is mean-zero Gaussian:

$$p(\mathbf{w} | \boldsymbol{\alpha}) = \prod_{i=0}^N N(w_i | 0, \alpha_i^{-1})^T \tag{8}$$

where $\boldsymbol{\alpha}$ -hyper parameters. This introduction of an individual hyper parameter for every weight is the key feature of the model, and is ultimately responsible for its sparsity properties. The posterior over the weights is then obtained from Bayes' rule:

$$p(\mathbf{w} | \mathbf{t}, \boldsymbol{\alpha}, \sigma^2) = (2\pi)^{\frac{-(N+1)}{2}} |\boldsymbol{\Sigma}|^{\frac{-1}{2}} \exp\left\{-\frac{1}{2}(\mathbf{w} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{w} - \boldsymbol{\mu})\right\}^T \tag{9}$$

with

$$\begin{aligned} \boldsymbol{\Sigma} &= (\boldsymbol{\Phi}^T \mathbf{B} \boldsymbol{\Phi} + \mathbf{A})^{-1} \\ \boldsymbol{\mu} &= \boldsymbol{\Sigma} \boldsymbol{\Phi}^T \mathbf{B} \mathbf{t} \end{aligned} \tag{10}$$

where $\mathbf{A} = \text{diag}(\alpha_0, \alpha_1, \dots, \alpha_N)$ and $\mathbf{B} = \sigma^{-2} \mathbf{I}_N$. Note that σ^2 is also treated as a hyper parameter, which may be estimated from the data.

By integrating out the weights, the marginal likelihood for the hyper parameters can be obtained:

$$p(\mathbf{t} | \boldsymbol{\alpha}, \sigma^2) = (2\pi)^{\frac{-N}{2}} |\mathbf{B}^{-1} + \boldsymbol{\Phi} \mathbf{A}^{-1} \boldsymbol{\Phi}^T|^{-\frac{1}{2}} \times \exp\left\{-\frac{1}{2} \mathbf{t}^T (\mathbf{B}^{-1} + \boldsymbol{\Phi} \mathbf{A}^{-1} \boldsymbol{\Phi}^T)^{-1} \mathbf{t}\right\} \tag{11}$$

The next step is to define hyper priors over $\boldsymbol{\alpha}, \sigma^2$ and integrate out the hyper parameters. However, the marginalization $\boldsymbol{\alpha}_{MP}, \sigma_{MP}^2$ can not be performed in closed form so that two alternative formulae for iterative re-estimation are used to solve the differential coefficient of (11):

$$\alpha_i^{new} = \gamma_i / \mu_i^2 \tag{12}$$

Where μ_i is the i -th posterior weights which can be computed from (10), and $\gamma_i = 1 - \alpha_i \sum_{ii}$, which can be interpreted as a measure of how ‘‘well-determined’’ each parameter w_i is. For the noise variance, both methods lead to the same re-estimate:

$$(\sigma^2)^{new} = \|\mathbf{t} - \boldsymbol{\Phi} \boldsymbol{\mu}\|^2 / (N - \sum_i \gamma_i) \tag{13}$$

The sparse Bayesian learning does not estimate from (12)-(13) and renew (9)-(10) iteratively until the convergence is obtained. During re-estimation, many of the $\alpha_i \rightarrow \infty$, which imply that the corresponding kernel functions can be pruned. When the convergence is achieved, according to the $\boldsymbol{\alpha}_{MP}, \sigma_{MP}^2$ and new data set \mathbf{x}^* , we can compute regression posterior distribution:

$$p(\mathbf{t}^* | \mathbf{t}, \boldsymbol{\alpha}_{MP}, \sigma_{MP}^2) = \mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\sigma}_*^2) \tag{14}$$

where $\boldsymbol{\sigma}_*^2 = \boldsymbol{\sigma}_{MP}^2 + \phi(\mathbf{x}^*)^T \boldsymbol{\Sigma} \phi(\mathbf{x}^*)$. So the nonlinear regression is:

$$y(\mathbf{x}^*; \mathbf{w}) = \boldsymbol{\mu}^* = \boldsymbol{\mu}^T \phi(\mathbf{x}^*) \tag{15}$$

So the RVM based on the sparse Bayesian learning could be treated with fewer relevance vectors than support vectors of SVM. Although the simulations show that the regression performance of the RVM is similar to the SVM, thanks to fewer relevance vectors, the estimation speed can be improved dramatically. Taking into account its nonlinear regression performance, the RVM can be employed for the nonlinear mapping for the identification of the inverse dynamics model.

2.3 Synthesis of Sub-networks Based on MRVM

According to the decomposition for the calculational subspaces for the inverse dynamics modeling, a synthesis RVM is offered to map the connection relation of subspaces $f_{i+1}(\cdot)$. So (2) is written as:

$$RVM_{i+1} : \hat{\mathbf{u}} = f_{i+1}(f_0(\mathbf{y}, \mathbf{u}^{(j)}, \mathbf{u}^{(j-1)}, \dots, \mathbf{u}^{(1)}), f_1(\mathbf{y}^{(1)}, \mathbf{u}^{(j)}, \mathbf{u}^{(j-1)}, \dots, \mathbf{u}^{(1)}), \dots, f_i(\mathbf{y}^{(i)}, \mathbf{u}^{(j)}, \mathbf{u}^{(j-1)}, \dots, \mathbf{u}^{(1)})) + \boldsymbol{\varepsilon}_{i+1} \tag{16}$$

In this way, a complicated high-dimensional model is simplified as $i+1$ low-dimensional sub-networks and one synthesis network. This modeling approach takes into consideration the influences of both time and spatial scales to the inverse dynamics of the nonlinear dynamic system. For each sub-network, parameters can be trained separately so that the morbid matrix and divergence problems for a high-dimensional model can be avoided. Fig. 1 shows the framework of MRVM based identification model for the inverse dynamics.

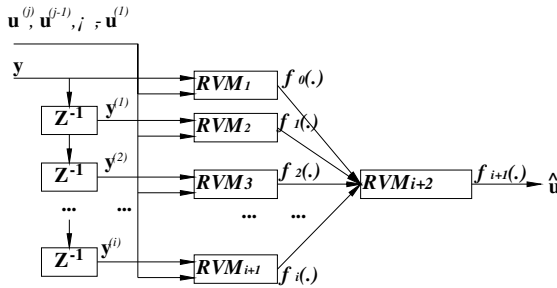


Fig. 1. The framework of MRVM based identification for the inverse dynamics model

3 The Identification Steps

Following steps show how to identify a MRVM based inverse dynamic model:

Step 1: Identification data acquisition. Acquire data continuously in accordance with the differential orders of the system. The given outputs is $(\mathbf{y}^{(i)}, \mathbf{y}^{(i-1)}, \dots, \mathbf{y}, \mathbf{u}^{(j)}, \mathbf{u}^{(j-1)}, \mathbf{u}^{(1)})$ and the inputs to be identified is \mathbf{u} . So a set of sample for identification is $\{\mathbf{u}; \mathbf{y}^{(i)}, \mathbf{y}^{(i-1)}, \dots, \mathbf{y}, \mathbf{u}^{(j)}, \mathbf{u}^{(j-1)}, \mathbf{u}^{(1)}\}$.

Step 2: Data segmentation. The next set of sample is $\{\mathbf{u}^{(-1)}; \mathbf{y}^{(i-1)}, \mathbf{y}^{(i-2)}, \dots, \mathbf{y}^{(-1)}, \mathbf{u}^{(j-1)}, \mathbf{u}^{(j-2)}, \mathbf{u}\}$. If the order recurs for $(m+n)$ times,

$(m+n)$ sets of sample can be obtained. In actual identifying processes the sampling frequencies of the inputs and outputs are not always the same. Fortunately, it will affect the sparsity of data block rather than the model results.

Step 3: Construct RVM sub-models using m -dimensional sample. Arranging samples in accordance with their orders, we can construct $i+1$ independent sub-networks RVM_k ($k \in [0, i]$) whose input is $(\mathbf{y}^{(k)}, \mathbf{u}^{(j)}, \mathbf{u}^{(j-1)}, \dots, \mathbf{u}^{(1)})$ and output \mathbf{u} .

Step 4: Construct the synthesis network to map the relations between different calculation subspaces. The rest n -dimensional samples are utilized by RVM_{i+1} to connect all subspaces so as to obtain the final identification model of the inverse dynamics. The former m -dimensional samples are not employed in this step in that different samples can improve the identifying capacity for different conditions.

Step 5: Test the trained nonlinear inverse dynamics model with testing samples. If the precision is acceptable, the nonlinear networks are employed to be the identification model which reflects the inversion of the system variables with different differential orders on input variables.

Having considered the dynamic variety of the nonlinear system, MRVM based approach employs the nonlinear regression network RVM and temporal-spatial transformation method to map functions $f_0(\cdot), f_1(\cdot), \dots, f_i(\cdot)$. Hence each sub-network has definite physical meanings. During the synthesis step of all sub-networks, RVM is also proposed to represent the dynamic function $f_{i+1}(\cdot)$. Comparing with linear synthesis methods, such as LS, Principal Component Analysis, RVM can reduce the regression error for the nonlinear problem to improve further identification precision for the inverse dynamics model.

4 Examples of the Inverse Dynamics Identification

There is a high temperature exchanger in a 600 MW supercritical generator that the steam temperature of the boiler fluctuates resulting from the variety of the spray flow rate of the cooler. It can be simplified as a SISO nonlinear system whose input u is the spray flow rate (kg/s) of the cooler, and the output y , the steam temperature ($^{\circ}\text{C}$). According to the mechanism analysis [10], when the load of the generator is 100%, the transfer function of the system is $-0.185/(1+18s)^2$.

The proposed approach is applied to model the inverse dynamics characteristics. Let the input be $u(t)=\sin(\pi t/80)+1.5\cos(\pi t/50)+2\sin(\pi t/30)$ so as to drive the system. In the identification process, let $\mathbf{x}=(y^{(-1)}, y^{(-2)}, u^{(1)})$ be the inputs of the inverse system and u be the inverse dynamics object to be identified. Set the sampling interval be 2s to acquire 300 sets of sample. 200 sets of sample are employed for sub-models training and the rest for synthesis RVM training. After constructing the MRVM based inverse dynamics identification model, randomly generate other 300 sets of testing sample, the furnished model is offered for the inverse dynamics identification. The identification results of the model are shown in Fig. 2. And the identification errors are shown in Fig. 3.

Root Mean Square Error (*RMSE*) and Maximal Absolute Error (*MAXAE*) are selected as error weight parameters. Computing from the above results, $RMSE=0.3188(\text{kg/s})$ and $MAXAE=0.9187(\text{kg/s})$, which shows that the proposed identification approach has better nonlinear inverting performance for the system.

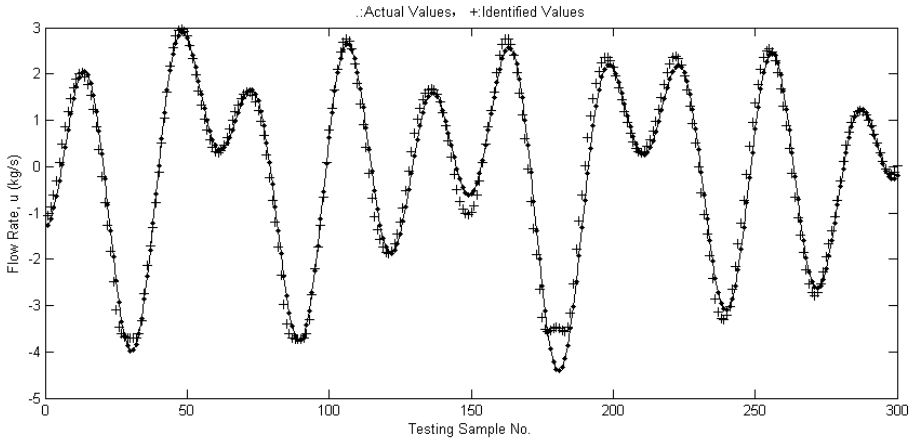


Fig. 2. Comparison between identified and actual results

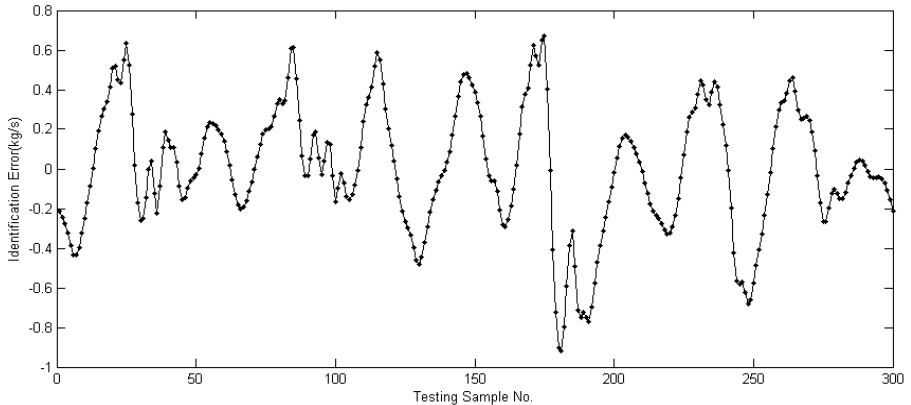


Fig. 3. Identification errors of the inverse dynamics model

5 Conclusions

The inputs of the system can be inverted from the historical input-output sequences. The inverse dynamics can be modeled from this process. In the paper a MRVM based identification approach of the inverse dynamics model was presented. According to their differential order for the system, the inputs and outputs were allocated into multiple calculational subspaces, RVMs. Another RVM was delivered to connect all subspaces. In this way, a complicated high-dimensional model is simplified as the synthesis of low-dimensional networks. This modeling approach takes into consideration the influences of both time and spatial scales to the inverse dynamics of the nonlinear system. A simulation example is put forward at last, the result shows that the proposed method has simple dynamic structure and clear physical meanings, which features in good

identification precision. The proposed modeling approach provides a novel tool for inverse dynamics identification.

Acknowledgments. This work is partially supported by Chongqing Key Technologies R&D Programme (2008AC3033) and Chongqing Municipal Education Commission Sci.&Tech. Programme (KJ080710).

References

1. Tan, S.H., Vandewalle, J.: Inversion of Singular System. *IEEE Trans. On Circuits Systems* 35(5), 583–587 (1988)
2. Dai, X.Z., He, D.: ANN Generalized Inversion for the Linearization and Decoupling Control of Nonlinear Systems. *IEEE Proc. Control Theory Appl.* 150(3), 267–277 (2003)
3. Singh, S.N.: Decoupling of Invertible Nonlinear Systems with State Feedback and Pre-compensation. *IEEE Trans. On Automatic Control* 24(6), 1237–1239 (1979)
4. Shen, S.G., Wang, G.J., Zhu, L.N.: Identification of Inverse Dynamics Model Based on Support Vector Machine and Its Application. *Journal of System Simulation* 20(1), 25–28 (2008)
5. Dai, X.Z., He, D., Zhang, X., et al.: MIMO System Invertibility and Decoupling Control Strategies Based on ANN α -th Order Inversion. *IEEE Proc. Control Theory Appl.* 148(2), 125–136 (2001)
6. Suykens, J.A.K., Tony, V.G., Jos, D.B.: *Least Squares Support Vector Machines*. K U Leuven, Belgium (2002)
7. Shaw, A.M., Doyle, F.J., Schwaber, J.S.: Dynamic Neural Network Approach to Nonlinear Process Modeling. *Computers and Chemical Engineering* 21(4), 371–385 (1997)
8. Tipping, M.T.: Sparse Bayesian Learning and the Relevance Vector Machine. *Journal of Machine Learning Research* 1(3), 211–244 (2001)
9. Faul, A.C., Tipping, M.E.: Analysis of Sparse Bayesian Learning. In: Dietterich, T.G., Becker, S., Ghahramani, Z. (eds.) *Advances in Neural Information Processing Systems*, vol. 14, pp. 383–389 (2002)
10. Deng, L.C., Wang, G.J., Chen, H.: On-line Fuzzy Identification of the Steam Temperature Object of Boiler. *Proceedings of the CSEE* 26(18), 111–115 (2006)

When Is Inconsistency Considered Harmful: Temporal Characterization of Knowledge Base Inconsistency

Du Zhang¹ and Hong Zhu²

¹ Department of Computer Science
California State University
Sacramento, CA 95819-6021, USA

zhangd@ecs.csus.edu
² Department of Computing
Oxford Brookes University
Oxford, Ox33 1HX, UK
hzhu@brookes.ac.uk

Abstract. Real world inconsistent information often has to do with not only what conflicting circumstances are but also when they happen. In this paper we present our research work on the temporal characteristics of inconsistent information that can exist in an intelligent system. To facilitate the discussions, we use knowledge base (KB) to refer to the component in an intelligent system that contains knowledge about a problem domain. Knowledge in a KB can be represented in terms of different formalisms, and plays a pivotal role in how an intelligent system accomplishes its intended tasks. The main results reported in this paper include: (1) establishing a formal definition for temporal inconsistency for knowledge in a KB in terms of the interval temporal logic; (2) describing a systematic approach to identifying conflicting intervals for temporally inconsistent assertions in a KB; and (3) delineating the semantic difference between the classical and temporal inconsistency.

Keywords: interval temporal logic, KB inconsistency, temporal inconsistency, conflicting intervals.

1 Introduction

Inconsistency in knowledge and information is ubiquitous in the real world and in the fields of computer science and artificial intelligence. Even though many of the conflicting cases we face are inconsequential, negligible, or unimportant. Sometimes the consequence of such inconsistent information can be grievous, detrimental, costly and devastating. When Ariane 5 launch vehicle was lost shortly after its lift off in 1996 due to a failed run-time check, the cause was traced back to the reuse of Ariane 4's software and the two had inconsistent launch trajectories [3]. The loss of the payload stood at more than 370 million dollars. Another high profile case was the Therac-25 accidents [12], one of the most widely cited software-related accidents in safety-critical systems. Therac-25 was a computerized radiation therapy machine. Software coding errors contributed to six known accidents where massive overdoses were involved by the Therac-25, with resultant deaths and serious injuries during the period from June 1985 to January 1987.

When inconsistency in a knowledge base (KB) results in two conflicting decisions such as “fire the spacecraft’s engine” and “do not fire the spacecraft’s engine,” a critical

issue is whether the two commands are issued at the same time or coincide in time. In general for two conflicting propositions, we are interested in knowing not only that they are inconsistent, but also when they become inconsistent. If the time intervals at which they each hold do not overlap, i.e., they do not occur simultaneously in the same time period, then there will not be a contradiction. To study the temporal properties of inconsistent information that exists in a decision making process, we need to resort to a temporal logic to formally establish temporal relationships between time periods of propositions and to reason about the presence or absence of temporal inconsistency with regard to a given KB. There are a number of temporal logic formalisms [1-2,4,7,14]. In this work, we choose to adopt James Allen's interval temporal logic [1-2], which is based on characterizing actions and events in terms of interval relationships.

Though there have been numerous studies on knowledge base inconsistency [5,8-11,13,15-19] and on temporal logic and its applications [1-2,4,7,14], respectively, the issue of characterizing KB inconsistency in terms of some temporal logic formalism has attracted little attention thus far as evidenced in the lack of published results in the literature.

The rest of the paper is organized as follows. Section 2 provides a brief overview on the interval temporal logic as was initially developed by James Allen. Section 3 summarizes the major types of KB inconsistency that will be reexamined under the temporal framework. Section 4 describes a formal definition for temporal inconsistency. How to identify conflicting intervals for temporal inconsistency is dealt with in Section 5. Finally, Section 6 concludes the paper with remarks on future work.

2 Interval Temporal Logic

The formalism we use in this work to delineate temporal inconsistency is based on James Allen's interval temporal logic (ITL) [1-2]. Using Emerson's criteria in [7], ITL can be considered as first-order, global, linear time, intervals, continuous and future operators. In this section, we provide a brief overview of interval temporal logic. For details, readers are encouraged to consult [1-2,4,7,14].

The basic temporal structure in ITL is a linear model of time. ITL starts with one primitive object, the time period or interval, and one primitive relation *Meets*. A time interval represents the time duration of some event that occurs or some property that holds in the world. Two intervals i and j meet if and only if i precedes j , but there is no time between i and j , and i and j do not overlap. There is a set of axioms about *Meets*: (1) every interval has an interval that meets it and another that it meets; (2) intervals can be concatenated to form a larger interval; (3) intervals uniquely define an equivalence class of intervals that meet them; (4) two intervals are equal if both meet the same interval and another interval meets them both; and (5) interval meeting spots can be ordered [2]. In addition, no interval can meet itself. If interval i meets interval j , then j cannot also meet i .

With the primitive object and the primitive relationship in place, the following list of additional interval relationships can be defined.

$\{ \textit{Before}(i, j), \textit{Overlaps}(i, j), \textit{Equals}(i, j), \textit{Starts}(i, j), \textit{During}(i, j), \textit{Finishes}(i, j) \}$

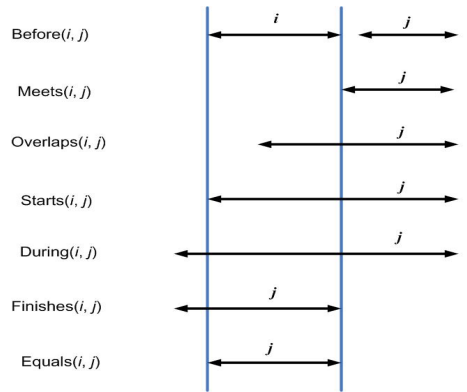


Fig. 1. Interval relationships for actions and events

Figure 1 offers a graphic illustration of the relationships. Definitions can be given for the inverse relationships of the aforementioned intervals: *MetBy*(j, i), *After*(j, i), *OverlappedBy*(j, i), *StartedBy*(j, i), *Contains*(j, i), and *FinishedBy*(j, i). Hence, there are thirteen interval relationships altogether.

Definition 1. For a given atomic formula $P(t_1, \dots, t_n)$ where P is the predicate symbol and t_1, \dots, t_n , are terms, we introduce a time period to the atom through adding two arguments \mathcal{S} , a time point, and \mathcal{D} , a duration: $P(t_1, \dots, t_n, \mathcal{S}, \mathcal{D})$. The semantic reading of the atom is that the truth value of $P(t_1, \dots, t_n)$ is *true* from the point of \mathcal{S} and for the period of \mathcal{D} . We can also use $\Delta = [\mathcal{S}, \mathcal{S} + \mathcal{D}]$ to represent the duration and Δ_p to denote the time interval for P .

3 Types of KB Inconsistency

There have been numerous studies on knowledge base inconsistency [5,8-11,13,15-19]. In this paper, we will focus the attention on our recent results in [19]. As described in [19], there are a number of different types of inconsistency that can exist in a knowledge base. Table 1 summarizes what has been defined in [19].

Table 1. Types of inconsistency

Inconsistency Type	Notation
$i\mathcal{I}_1$: Complementary	$L_i \neq L_j$
$i\mathcal{I}_2$: Mutually exclusive	$L_i \not\subseteq L_j$
$i\mathcal{I}_3$: Incompatible	$L_i \not\supseteq L_j$
$i\mathcal{I}_4$: Anti-subtype	$L_i \not\sqsubseteq L_j$
$i\mathcal{I}_5$: Asymmetric	$L_i \not\supseteq L_j$
$i\mathcal{I}_6$: Anti-inverse	$L_i \neq L_j$
$i\mathcal{I}_7$: Disagreeing	$L_i \not\supseteq L_j$
$i\mathcal{I}_8$: Contradictory	$L_i \not\supseteq L_j$

Here are brief explanations to the aforementioned types of inconsistency.

i₁. Given A_1 and A_2 as syntactically identical atoms (same predicate symbol, same arity, and same terms at corresponding positions), A_1 and $\neg A_2$ (or $\neg A_1$ and A_2) are referred to as *complementary literals*. We denote complementary literals as $L_1 \neq L_2$ where L_1 and L_2 are an atom and its negation.

i₂. Given two literals L_1 and L_2 that are syntactically different and semantically opposite of each other (the assertion of L_1 (L_2) implies the negation of the other L_2 (L_1)), we call L_1 and L_2 as *mutually exclusive literals* and use $L_1 \neq L_2$ to denote it.

i₃. For two literals L_1 and L_2 that are syntactically different, but logically equivalent, we call them *synonymous*, and denoted $L_1 \cong L_2$. For instance, the following are synonymous: *Father*(x , john) \cong *Male_parent*(x , john). Given L_1 and L_2 that are a complementary pair of synonymous literals, we call them *incompatible literals*, and denote $L_1 \neq L_2$.

i₄. For two literals (or concepts as in description logics) L_1 and L_2 , if L_1 is a *subtype* or a *specialization* of L_2 , then we say that L_1 is subsumed by L_2 (or L_2 subsumes L_1) and use $L_1 \sqsubseteq L_2$ to denote that. For example, *Surgeon*(john) \sqsubseteq *Doctor*(john). If L_1 is no longer a subtype of L_2 , then we call L_1 and L_2 *anti-subtype literals* and use the following to denote it: $L_1 \not\sqsubseteq L_2$.

i₅. Given two literals L_1 and L_2 that share the same predicate, if the predicate in L_1 and L_2 is a *symmetric* relation (i.e., if L_1 is $p(x, y)$ and L_2 is $p(y, x)$, then we have $\forall x, y [p(x, y) \supset p(y, x)]$), L_1 and L_2 are referred to as *symmetric* and are denoted as $L_1 \sphericalangle L_2$. When the predicate in L_1 and L_2 is a symmetric relation, but L_1 and L_2 are no longer symmetric literals, we say that L_1 and L_2 are *asymmetric* and use $L_1 \not\sqsupset L_2$ to denote that.

i₆. When two predicates represent relationships that are inverse of each other, we call them *inverse predicates*. Literals L_1 and L_2 , are referred to as *inverse literals* when they contain inverse predicates and represent inverted relationships. We use $L_1 \simeq L_2$ to denote that. When predicates in L_1 and L_2 represent inverse relationships but L_1 and L_2 are no longer inverse literals, we say that L_1 and L_2 are *anti-inverse* and denote it with $L_1 \neq L_2$.

i₇. Given L_1 and L_2 for the same proposition and L_2 is at a more concrete level of abstraction than L_1 , we call them *reified literals* and denote it with $L_1 \cong L_2$. If reified quantities in L_1 and L_2 are no longer compatible, we say that L_1 and L_2 are *disagreeing* and use $L_1 \not\cong L_2$ to denote it.

i₈. Given literals L_1 and L_2 with either the same or different predicate symbols, if they contain attributes (terms) which violate type restrictions or integrity constraints, we refer to L_1 and L_2 as *contradictory* and denote it with $L_1 \neq L_2$.

To facilitate the temporal characterization of the aforementioned inconsistency, we assume that all the predicates include the two temporal arguments as specified in Definition 1.

4 Temporal Inconsistency

Definition 2. The time periods of two formulas are *coinciding* when there exists one of the following interval relationships between the two: *Overlaps*, *OverlappedBy*, *Starts*, *StartedBy*, *During*, *Contains*, *Finishes*, *FinishedBy*, or *Equals*. We use the

predicate $\mathcal{CP}(\Delta_P, \Delta_Q)$ to denote the existence of the coinciding period between formulas P and Q, and $\neg\mathcal{CP}(\Delta_P, \Delta_Q)$ to indicate that P and Q are non-coinciding.

$$\begin{aligned} \mathcal{CP}(\Delta_P, \Delta_Q) \equiv_{\text{def}} & [Overlaps(\Delta_P, \Delta_Q) \vee Starts(\Delta_P, \Delta_Q) \vee During(\Delta_P, \Delta_Q) \\ & \vee Finishes(\Delta_P, \Delta_Q) \vee Equals(\Delta_P, \Delta_Q) \vee OverlappedBy(\Delta_Q, \Delta_P) \\ & \vee StartedBy(\Delta_Q, \Delta_P) \vee Contains(\Delta_Q, \Delta_P) \vee FinishedBy(\Delta_Q, \Delta_P)]. \end{aligned}$$

Definition 3. Given two literals L_1 and L_2 , if they are *consistent*, then they are denoted as $\models(L_i, L_j)$. We say that L_1 and L_2 are *conflict* literals, denoted as $\not\models(L_i, L_j)$, if we have the following:

$$\begin{aligned} \not\models(L_i, L_j) \equiv_{\text{def}} & [(L_i \neq L_j) \vee (L_i \neq L_j) \vee (L_i \neq L_j) \vee (L_i \not\subseteq L_j) \\ & \vee (L_i \not\supseteq L_j) \vee (L_i \neq L_j) \vee (L_i \not\supseteq L_j) \vee (L_i \neq L_j)]. \end{aligned}$$

Definition 4. Given a set Ω of literals, Ω is *non-conflicting*, denoted as $\models\Omega$, if

$$\forall L_i, L_j \in \Omega [\models(L_i, L_j)].$$

Ω is *conflicting*, denoted as $\not\models\Omega$, if $\forall L_i \in \Omega \exists L_j \in \Omega [\not\models(L_i, L_j)]$.

Now, we can accurately define what temporal inconsistency entails.

Definition 5. Given two literals L_i and L_j , *temporal inconsistency*, denoted as $\not\models t$, between L_i and L_j can be formally defined as follows:

$$\not\models t(L_i, L_j) \equiv_{\text{def}} [\not\models(L_i, L_j) \wedge \mathcal{CP}(\Delta_{L_i}, \Delta_{L_j})].$$

Thus, L_i and L_j are said to be *temporal inconsistent* if L_i and L_j are conflicting and their time intervals are coinciding.

Given L_i and L_j , let $\Delta = \Delta_{L_i} \cap \Delta_{L_j}$ denote the largest sub-interval that is contained in both Δ_{L_i} and Δ_{L_j} . Thus, Δ is the maximum coinciding interval for L_i and L_j when we have $\mathcal{CP}(\Delta_{L_i}, \Delta_{L_j})$. For $\not\models t(L_i, L_j)$, when we have to be specific about the interval over which temporal inconsistency takes place, we use the following notation to indicate that L_i and L_j are temporally inconsistent over Δ : $\not\models t(L_i, L_j)_\Delta$.

Definition 6. *Temporal consistency*, denoted as $\models t(L_i, L_j)$ can be defined as

$$\models t(L_i, L_j) \equiv_{\text{def}} [(\models(L_i, L_j) \wedge Equals(\Delta_{L_i}, \Delta_{L_j})) \vee (\not\models(L_i, L_j) \wedge \neg\mathcal{CP}(\Delta_{L_i}, \Delta_{L_j}))].$$

Hence, two literals L_i and L_j are *temporal consistent* if they are consistent and have the same time interval¹, or if they are conflicting but non-coinciding. Similarly, we can use $\models t(L_i, L_j)_\Delta$ to denote that L_i , and L_j are temporally consistent over Δ ².

Definition 7. Given $\not\models t(L_i, L_j)$, depending on the interval relationship between the two literals, we have the following types of temporal inconsistency:

- *Congruent*: denoted as $\not\models t(L_i, L_j)_\Delta^c$, if $[\not\models(L_i, L_j) \wedge Equals(\Delta_{L_i}, \Delta_{L_j})]$;
- *Subsuming*: denoted as $\not\models t(L_i, L_j)_\Delta^s$, if

¹ The reason that we insist on the consistent literals having the same time interval is due to the fact that any other interval relationship between the two would result in them having opposite truth values during a sub-interval in either Δ_{L_i} or Δ_{L_j} .

² When $\neg\mathcal{CP}(\Delta_{L_i}, \Delta_{L_j})$, then Δ in $\models t(L_i, L_j)_\Delta$ may represent two disjoint time intervals.

$$[\neq_t(L_i, L_j) \wedge (\text{Starts}(\Delta_{L_i}, \Delta_{L_j}) \vee \text{During}(\Delta_{L_i}, \Delta_{L_j}) \vee \text{Finishes}(\Delta_{L_i}, \Delta_{L_j}) \\ \vee \text{StartedBy}(\Delta_{L_i}, \Delta_{L_j}) \vee \text{Contains}(\Delta_{L_i}, \Delta_{L_j}) \vee \text{FinishedBy}(\Delta_{L_i}, \Delta_{L_j}))];$$

- *Overlapping*: denoted as $\neq_t(L_i, L_j)^\circ_\Delta$, if $[\neq_t(L_i, L_j) \wedge (\text{Overlaps}(\Delta_{L_i}, \Delta_{L_j}) \vee \text{OverlappedBy}(\Delta_{L_j}, \Delta_{L_i}))]$.

For the congruent case, temporal inconsistency for both literals will be *persistent*. For the subsuming case, the literal (L_i) whose interval is subsumed by that of the other will have a persistent temporal inconsistency. In the overlapping case, there is a chance for both literals to be still temporal consistent.

Given a set of intervals $\{\Delta_i, \dots, \Delta_k\}$, we use the following to indicate the concatenation of intervals in the set in the chronological order: $\bigwedge \{\Delta_i, \dots, \Delta_k\}$. The result is an interval, which we assume contains no gap.

Definition 8. Given a set Ω of literals and $L_i \in \Omega$, L_i is *completely temporal inconsistent*, denoted as $\text{CTI}(L_i)$, if either of the following holds:

- $\exists L_j \in \Omega [(\neq_t(L_i, L_j)^\circ_\Delta) \vee (\neq_t(L_i, L_j)^s_\Delta)]$;
- $\exists L_j, \dots, L_k \in \Omega [(\neq_t(L_i, L_j)^\circ_\Delta \wedge \dots \wedge \neq_t(L_i, L_k)^\circ_\Delta) \\ \wedge (\Delta_{L_i} = \bigwedge \{(\Delta_{L_i} \cap \Delta_{L_j}), \dots, (\Delta_{L_i} \cap \Delta_{L_k})\})]$.

Definition 9. Given a set Ω of literals and $L_i \in \Omega$, L_i is *partially temporal inconsistent*, denoted as $\text{PTI}(L_i)$, if the following holds:

$$\exists L_j \in \Omega \exists \Delta' \subset \Delta_{L_i} \neg \exists L_k \in \Omega [\neq_t(L_i, L_j) \wedge \neq_t(L_i, L_k)_{\Delta'}].$$

Definition 10. Given a set Ω of literals and $L_i \in \Omega$, L_i is *fully temporal consistent*, denoted as $\text{FTC}(L_i)$, if the following holds: $\forall L_j \in \Omega [\neq_t(L_i, L_j)]$.

5 When Does Inconsistency Become Harmful

Once conflicting literals are identified in a KB, we need to ascertain whether their time intervals are coinciding to determine if they constitute temporal inconsistency. If two literals are temporally inconsistent, we want to identify the coinciding or conflict interval between the two. Table 2 summarizes the conditions to be used to identify specific temporal relations and the resultant conflict intervals with regard to different temporal relationships. Once we obtain the conflict intervals for all temporal inconsistent cases in a KB, we are in a position to characterize the temporal properties for the entire KB.

Definition 11. Given two literals L_i and L_j with their respective time intervals Δ_{L_i} , Δ_{L_j} , if $\neq_t(L_i, L_j)$ and $\mathcal{AP}(\Delta_{L_i}, \Delta_{L_j})$, then their conflicting interval can be obtained according to Table 2.

Example 1. For the following pair of literals with the last two arguments indicating the \mathcal{S} and \mathcal{D} values, respectively:

Send_msg_to(agent1, agent2, 5, 10), -Received_msg_from(agent2, agent1, 6, 12), this conflict case is of $i\mathcal{S}_7$ and there exists a temporal relation of $t\mathcal{R}_{11}$ between the two literals with a conflict interval of $\Delta = [6, 10]$.

Table 2. Pair-wised conflict intervals for $i\mathcal{F}_1 - i\mathcal{F}_8$

Temporal Relationship	Conditions	Conflict Interval
$t\mathcal{R}_1: Before(\Delta_{L_i}, \Delta_{L_j})$	$(\Delta_{L_i} < \Delta') \wedge (\Delta' < \Delta_{L_j})^3$	\emptyset
$t\mathcal{R}_2: After(\Delta_{L_j}, \Delta_{L_i})$	$(\Delta_{L_i} < \Delta') \wedge (\Delta' < \Delta_{L_j})$	\emptyset
$t\mathcal{R}_3: Meets(\Delta_{L_i}, \Delta_{L_j})$	$(\Delta_{L_i} < \Delta_{L_j})$	\emptyset
$t\mathcal{R}_4: MetBy(\Delta_{L_j}, \Delta_{L_i})$	$(\Delta_{L_j} < \Delta_{L_i})$	\emptyset
$t\mathcal{R}_5: Starts(\Delta_{L_i}, \Delta_{L_j})$	$(\mathcal{S}_i = \mathcal{S}_j) \wedge (\mathcal{Q}_i + k = \mathcal{Q}_j) \wedge (0 < k)$	Δ_{L_i}
$t\mathcal{R}_6: StartedBy(\Delta_{L_j}, \Delta_{L_i})$	$(\mathcal{S}_j = \mathcal{S}_i) \wedge (\mathcal{Q}_i + k = \mathcal{Q}_j) \wedge (0 < k)$	Δ_{L_j}
$t\mathcal{R}_7: During(\Delta_{L_i}, \Delta_{L_j})$	$(\mathcal{S}_j < \mathcal{S}_i) \wedge (\mathcal{Q}_i < \mathcal{Q}_j)$	Δ_{L_i}
$t\mathcal{R}_8: Contains(\Delta_{L_j}, \Delta_{L_i})$	$(\mathcal{S}_i < \mathcal{S}_j) \wedge (\mathcal{Q}_i < \mathcal{Q}_j)$	Δ_{L_j}
$t\mathcal{R}_9: Finishes(\Delta_{L_i}, \Delta_{L_j})$	$(\mathcal{S}_i = \mathcal{S}_j + k) \wedge (\mathcal{Q}_i + k = \mathcal{Q}_j) \wedge (0 < k)$	Δ_{L_i}
$t\mathcal{R}_{10}: FinishedBy(\Delta_{L_j}, \Delta_{L_i})$	$(\mathcal{S}_j = \mathcal{S}_i + k) \wedge (\mathcal{Q}_i + k = \mathcal{Q}_j) \wedge (0 < k)$	Δ_{L_j}
$t\mathcal{R}_{11}: Overlaps(\Delta_{L_i}, \Delta_{L_j})$	$(\mathcal{S}_j = \mathcal{S}_i + k) \wedge (\mathcal{S}_j < \mathcal{S}_i + \mathcal{Q}_i) \wedge ((\mathcal{S}_i + \mathcal{Q}_i - k) < \mathcal{Q}_j) \wedge (0 < k < \mathcal{Q}_j)$	$[\mathcal{S}_i + k, \mathcal{S}_i + \mathcal{Q}_i]$
$t\mathcal{R}_{12}: OverlappedBy(\Delta_{L_j}, \Delta_{L_i})$	$(\mathcal{S}_i = \mathcal{S}_j + k) \wedge (\mathcal{S}_i < \mathcal{S}_j + \mathcal{Q}_j) \wedge ((\mathcal{S}_j + \mathcal{Q}_j - k) < \mathcal{Q}_i) \wedge (0 < k < \mathcal{Q}_j)$	$[\mathcal{S}_j + k, \mathcal{S}_j + \mathcal{Q}_j]$
$t\mathcal{R}_{13}: Equals(\Delta_{L_i}, \Delta_{L_j})$	$(\mathcal{S}_i = \mathcal{S}_j) \wedge (\mathcal{Q}_i = \mathcal{Q}_j)$	Δ_{L_i} or Δ_{L_j}

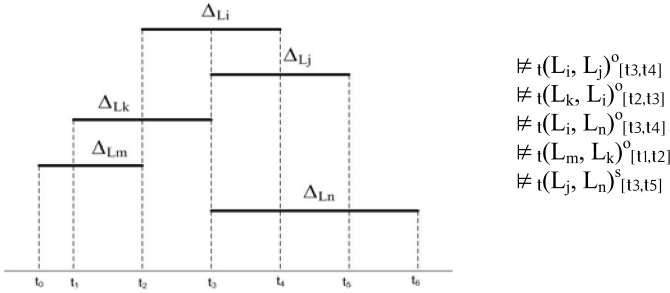


Fig. 2. Intervals of literals in Ω .

Example 2. If we have the following literals

Dispatched_to(agent1, host2, **3, 10**), *Dispatched_to*(agent1, host3, **3, 15**),

this is an $i\mathcal{F}_2$ case with $t\mathcal{R}_5$ temporal relation. It is a subsuming case.

Example 3. Given a set Ω of literals $\{L_i, L_j, L_k, L_m, L_n\}$, $\# \Omega$. Figure 2 indicates the intervals for the literals in Ω .

For the five literals in Ω , we have: $CTI(L_i)$, $CTI(L_j)$, $CTI(L_k)$, $PTI(L_m)$, and $PTI(L_n)$.

6 Related Work and Concluding Remarks

The work reported in [6] deals with the use of a paraconsistent temporal logic called QCTL for verifying temporal properties of inconsistent concurrent systems. In QCTL, temporal operators (next, eventually, always, and until) and path quantifiers (there

³ Δ' is a non-empty interval for $t\mathcal{R}_1$ and $t\mathcal{R}_2$ in Table 2.

exists a path, for all paths) are utilized to represent and reason about inconsistent specifications in concurrent systems. A model checking technique was proposed to verify the temporal properties of inconsistent concurrent systems over QCTL. There was no discussion on the types of inconsistency the approach can deal with.

In this paper, we describe a novel approach in characterizing the temporal aspects of KB inconsistency. The formalism utilized in our work is interval temporal logic. We provide a formal definition for temporal inconsistency that recognizes the fact that there are two important underpinnings for the real world inconsistent knowledge: logically conflicting and temporally coinciding. We also describe how to identify conflicting intervals for temporally inconsistent statements in a KB and delineated the semantic difference between the classical and temporal inconsistency. Our contribution lies in the fact that the reported results fill a gap in the temporal characterization of KB inconsistency.

Future work can be pursued in developing tools that can be used to identify different types of temporal inconsistency.

Acknowledgments. The authors would like to acknowledge the comments from anonymous reviewers.

References

1. Allen, J.F.: Toward a General Theory of Action and Time. *Artificial Intelligence* 23(2), 123–154 (1984)
2. Allen, J.F., Ferguson, G.: Actions and Events in Interval Temporal Logic. *Journal of Logic and Computation* 4(5), 531–579 (1994)
3. Ariane 5 Flight 501 Failure Full Report, <http://sunnyday.mit.edu/accidents/Ariane5accidentreport.html>
4. Bennett, B., Galton, A.: A Unifying Semantics for Time and Events. *Artificial Intelligence* 153(1-2), 13–48 (2004)
5. Brachman, R.J., Levesque, H.J.: *Knowledge Representation and Reasoning*. Morgan Kaufmann Publishers, San Francisco (2004)
6. Chen, D., Wu, J.: Model Checking temporal Aspects of Inconsistent Concurrent Systems based on Paraconsistent Logic. *Electronic Notes in Theoretical Computer Science*, vol. 157, pp. 23–38 (2006)
7. Emerson, E.A.: Temporal and Modal Logic. In: van Leeuwen, J. (ed.) *Handbook of Theoretical Computer Science*, North-Holland Pub. Co., Amsterdam (1995)
8. Genesereth, M.R., Nilsson, N.J.: *Logical Foundations of Artificial Intelligence*. Morgan Kaufmann Publishers, Inc., Los Altos (1987)
9. Grant, J., Hunter, A.: Measuring Inconsistency in Knowledge Bases. *Journal of Intelligent Information Systems* 27, 159–184 (2006)
10. Hunter, A., Konieczny, S.: Approaches to Measuring Inconsistent Information. In: Bertossi, L., Hunter, A., Schaub, T. (eds.) *Inconsistency Tolerance*. LNCS, vol. 3300, pp. 189–234. Springer, Heidelberg (2005)
11. Knight, K.: *A Theory of Inconsistency*, Ph.D. Dissertation, Department of Mathematics, the University of Manchester, UK (2002)
12. Leveson, N., Turner, C.S.: An Investigation of the Therac-25 Accidents. *IEEE Computer* 26(7), 18–41 (1993)

13. Levesque, H.J., Lakemeyer, G.: *The Logic of Knowledge Bases*. The MIT Press, Cambridge (2000)
14. Manna, Z., Pnueli, A.: *The Temporal Logic of Reactive and Concurrent Systems: Specification*. Springer, New York (1992)
15. Rushby, J., Whitehurst, R.A.: *Formal Verification of AI Software*. NASA Contractor Report 181827 (February 1989)
16. Zhang, D., Nguyen, D.: PREPARE: A Tool for Knowledge Base Verification. *IEEE Transactions on Knowledge and Data Engineering* 6(6), 983–989 (1994)
17. Zhang, D., Luqi: Approximate Declarative Semantics for Rule Base Anomalies. *Knowledge-Based Systems* 12(7), 341–353 (1999)
18. Zhang, D.: Fixpoint Semantics for Rule Base Anomalies. *International Journal of Cognitive Informatics and Natural Intelligence* 1(4), 14–25 (2007)
19. Zhang, D.: *On Classifying Inconsistency in Autonomic Agent Systems*, Technical Report, Department of Computer Science, California State University, Sacramento (submitted for publication) (December 2007)

Intelligent Engineering and Its Application in Policy Simulation

Xiaoyou Jiao¹ and Zhaoguang Hu²

¹ School of Electrical Engineering, Beijing Jiaotong University, Beijing 100044, China

² State Power Economic Research Institute of SG, Beijing 100761, China
{jiaoxiaoyou, huzhaoguang}@chinasperi.sgcc.com.cn

Abstract. The balance of electric power supply and demand is the important precondition of the sustainable development of power industry and national economy. In China, the government adjusts industry structure by putting macro-policy in practice for the balance. But the relationship between macro-policy and electric power supply and demand is non-linear and complicated, namely that policy simulation is a semi-structure problem, and many non-linear relationships and uncertain factors can't be simulated in traditional linear model. Intelligent Engineering (IE) is a kind of methodology, System Engineering (SE) offers direction for perfecting and applying IE, This paper focuses on the combination of Distributed Artificial Intelligence (DAI) and Cybernetics, and does an innovative research about Distributed Intelligent Control (DIC). Based on IE and optimized solution theory, the controllability of DIC system are defined for the first time. As a sample, agent-based intelligent simulation system is built to simulate the influence from macro policy to electric power supply and demand.

Keywords: Balance of electric power supply and demand; Policy Simulation; Intelligent Engineering (IE); Distributed Intelligent Control (DIC); Agent-based intelligent simulation system.

1 Introduction

The traditional Cybernetics has begun to solve some social or economy control problems. Based on the precise mathematics matrix, many models and control theories have been studied [1,2,3]. All of these models simplify the complicated system as a linear system, and build a precise quantity relationship by a series of linear simultaneous equations among various factors. However, for policy simulation, the conception of macro-policy is abstract and its macro-effect is a dynamic process, which is a semi-structure problem, so it's a unique difficulty to build a precise and quantitative mathematics model based on traditional method from macro-policy to electric power supply and demand.

With the rapid development of Artificial Intelligence (AI), Neural Net (NN) and Fuzzy System (FS), it's necessary and advanced to combine Intelligence Theory with Cybernetics to solve the complicated system problems [4,5,6]. During these years,

Multi-agent technology and Distributed Artificial Intelligence (DAI) offer new approach to intelligence control, it doesn't emphasize on the system characteristics about single, enormous and complicated, but divides the system into various agents by its function, and every agent can communicate and harmonize with each other to accomplish the complicated system control tasks together. So, in view of the difficulties in policy simulation, we combined Intelligent Engineering (IE) [7, 8], Cybernetics and Agent technology to build a frame about Distributed Intelligent Control (DIC) system, and set up a IE platform to simulate the macro-policy effect.

2 DIC System Based IE

The appearance of masses of non-linear phenomena and small probability events leads up to the system complexity research, and this is also the meaning of intelligence science, So DIC system is constructed. The main research achievements of some organizations and scholars in this domain are summarized as table 1.

Table 1. Comparative Table about DIC

Related Examples	Feature
DVMT	Abstract and distributed system model
MACE	Collateral DAI system
HECODES	Uncertain collaboration and problem solving
DKPS	Logic-knowledge model and knowledge share

It can be seen that with DIC theory to solve great system problems is a new and effective way, so we combine IE with Cybernetics and Agent technology, IE should not only combine with AI, NN, FS but also should take human knowledge into consideration. In this paper, the DIC system adopts the parallel processing structure in DAI and feedback adjustment in Cybernetics, in the guidance of Complex System Theory, three different intelligence form and hierarchy about machine intelligence, human intelligence and non- intelligence are integrated to solve the complicated, distributed and semi-structure system problems. The existing MAS system have had the ability to make agents to communicate with each other, but the agent itself has not been provided with illation function. In this paper, Rough Set theory (RS) and fuzzy illation method are applied to actualize the rules mining from macro-policy to electric power supply and demand, and then the agent is endowed with illation ability, consequently, MAS will be an absolute intelligent system: intelligent, distributed, interactive.

Definition1. DIC system is described by the following six tuple set:

$$\{A_n, C_m, C_p, C_d, R_u, B_a\},$$

Where, intelligence object (IO) is: $A = \langle A_n, R_u, B_a \rangle$. (1)

Intelligence framework is: $T = \{C_m, C_p, C_d\}$. (2)

A_n is every kind of system element; R_u is forward-illation function of IO, namely positive feedback; B_a is back-illation function of IO, namely negative feedback; T describes the correlation among IO, C_m is communication mechanism; C_p is

cooperation mechanism; C_d is coordination mechanism. Game analysis will be done based on the three mechanisms.

Definition2. In DIC system, the relationship equation among output variable y , state variable u and input variable x is:

$$y = I(x_1, x_2, \dots, x_n; u_1, u_2, \dots, u_n). \tag{3}$$

Where I is intelligence operator in IE theory, it includes AI operator, NN operator, FS operator and so on.

In this paper, an economy circulation exists in the policy simulation: macro-policy impacts investment, consumption, import, export and the adjustment of industry structure, consequentially the latter will result in the change of electric power consumption, afterward, decision-makers will adjust the policy based on the change. As shown in Fig1, output variable is electric power vector, and input variable is macro-policy vector, state variable is investment, consumption, import and export vectors.

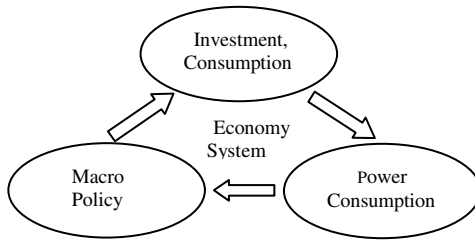


Fig. 1. Circulation in Policy Simulation

Researching the controllability of DIC system is significant for problem solving by IE. The strict controllability criterion exists in linear time-invariant system, however, the control from macro-policy to electric power demand and supply is a non-linear complex system, it's difficult to set an exact and sufficient quantized model. So based on IE and α -optimized solution theory, the controllability of DIC system is defined in the paper.

In IE theory, intelligent path is a set of fuzzy relations, mappings, transformations and all the ways between the initial state set S_0 and target state set D . It can be in the form as:

$$P : S_0 \rightarrow D \tag{4}$$

Intelligent space is defined as: $I = \langle P, S \rangle$, where, S is the set of states and P is the set of intelligent path between S_0 and D , S_0 and D are subset of S .

Definition3. An α -optimized solution for the problem $B = \langle S_0, D, PB \rangle$ is defined as $SL(\alpha)$, if there is a fuzzy set fP in intelligent paths set PB

$$fP : \rightarrow PB[0,1] \tag{5}$$

And, $SL(\alpha) = \{x \mid \mu_{fP}(x) \geq \alpha, x \in PB\} \tag{6}$

Where, $\beta \in [0,1]$.

Definition4. DIC system has the characteristics of intelligence controllability, if in intelligent space $\langle P, S \rangle$, $SL(\alpha)$ is the α - optimized solution for the problem $B = \langle S_0, D, PB \rangle$, for $\forall \varepsilon > 0$, if there is $p_t \in PB$, and

$$\left\| \mu_{p_t}(pt) - \alpha \right\| < \varepsilon \tag{7}$$

Compared with the traditional Cybernetics whose control requirement is that system must arrive at a fixed state at terminal time T , intelligence controllability allows that the system states arrive in an area $B(\ker(SL_\alpha), \varepsilon)$ at terminal time T ($B(\ker(SL_\alpha), \varepsilon)$ which is an open ball whose spherical center is $\ker(SL_\alpha)$, the radius is ε , and $\ker(SL_\alpha)$ is such element whose membership is “1” in $SL(\alpha)$). The significance of intelligence controllability is that system states can be intelligently controlled by system input through intelligence operator, consequently anticipative dynamic output can be acquired.

In a similar way, the definition of observability of DIC system can be got.

3 The Agent Model Based on Rules

Because of the nonlinearity and uncertainty of policy simulation, to set up a nonlinear intelligent mathematic model is the most important thing for applying the DIC. The rule-based models are set up with the ideas in IE theory, and then the models can endow the agent with the ability of intelligent illation.

Defination7. System input is translated into output in DIC system by intelligence operator:

$$f : X \rightarrow Y \tag{8}$$

And its mathematic model is defined as:

$$f(X, Y) = 0 \tag{9}$$

Where, X, Y represent two different states of nonlinear complicated systems, in the DIC systems, X is input variable, Y is output variable, f is the nonlinear arithmetic based on rules between input and output, which contains all kinds of different intelligence operators, and it is called the rule-based models.

The intelligent problem-solving based on rules is a process where knowledge information (fact, existing state or environment background, etc) are input into the DIC system, then anticipant target state is gained by rules matching, logic illation and other relevant intelligence operator.

The expression form of logic rules is:

If A then B , where $A \in X, B \in Y$.

The whole DIC system based on multi-agent, as shown in Fig2, includes financial policy agent, monetary policy agent, task decomposition and distribution agent, rules agent, result integration agent, effect evaluation agent, control agent, electric power supply and demand agent, apperceive agent, etc.

In the feedback part of system simulation, e.g. suppose the electric power demand is high under the effect of initial policy, so it needs to adjust the policy through feedback. Here the rational optimization path must not be oriented to the direction of

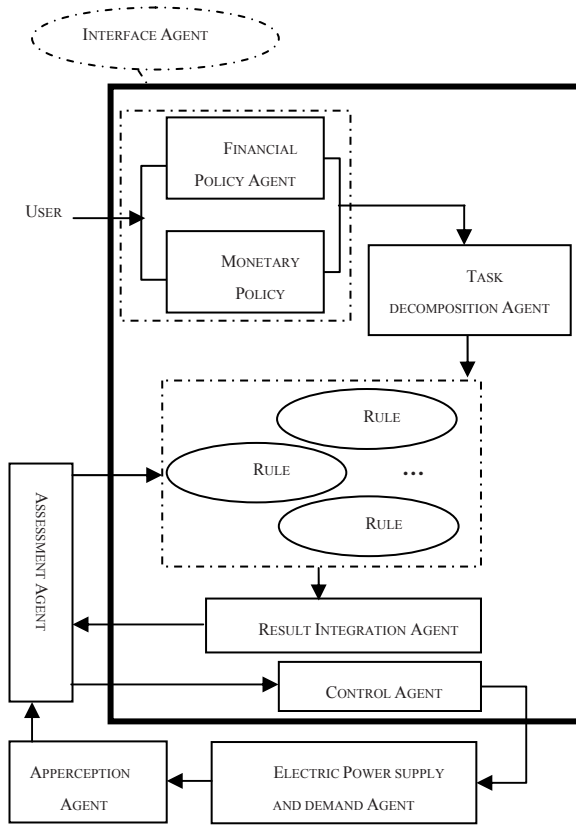


Fig. 2. System Frame of Multi-agent Intelligent Simulation

making a higher demand but the one which make it lower. So, we set a special “human-computer cooperation optimization” structure in this part: introducing the human qualitative thinking to guide the establishment of the initial value and the chief direction of optimization. This part is comprised of artificial optimization agent and intelligent optimization agent. The main idea of agent design is illustrated in Fig3.

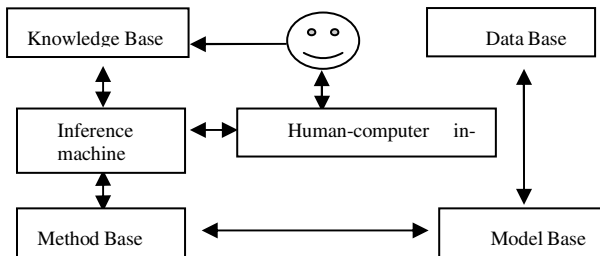


Fig. 3. Infrastructure of Agent Design Method

The knowledge base is policy rules for agent inference. The inference machine is based on fuzzy logic to implement the reasoning process. The method base manages a set of methods such as problem-solving method and optimization method, while model base is subordinated to method base and manage sets of concrete models. Finally the database manages and stores the data and information for the agent.

In Fig4, it shows the inner configuration of the agent[9], First, it contains state request and state transition function to manage the agent states which include Initiated, Active, Suspended, Waiting, Deleted and Transit according to the FIPA definition. When the macro-policy changes, the agent will begin to initialize or end to work. All agents' structure is distributed, and between these agents they have the ability to communicate and transmit information, which will help them to associate to solve the problems by the intelligent control.

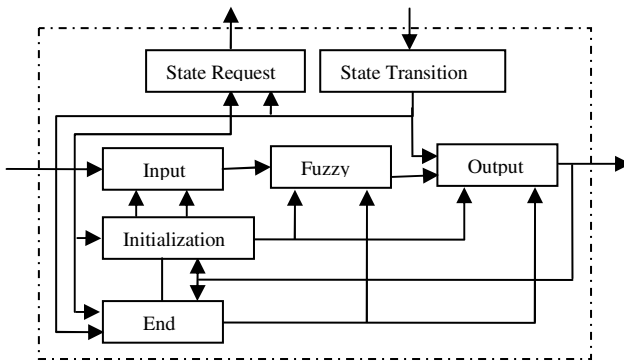


Fig. 4. Inner Configuration of the Agent

4 Rules Mining and Fuzzy Illation

This paper will manage to mine the rules based on history data (1990-2005), however, in view of imperfection and imprecision, it is also difficult to mine rules in these data. Rough Set (RS) is a mathematic tool to deal with the uncertainty which put forward by Z. Pawlak in 1982 [10].

The steps about rules mining based on the RS and Fuzzy -illation include:

1) Pretreatment of historical data: the thing regarded mostly by the rough set theory is separate distributed attribute space, so at first it is necessary to make attribute discretization to sample data [11]; monetary policy includes lending rate, reserve ratio, money supply M2. Financial policy includes government expenditure, revenue and net export, electric power data are obtained from China Energy Statistical Yearbook.

2) Setting up and reduction of the decision table: Dividing the attribute value into some intervals based on membership function, every interval is figured with different codes, using the worked condition attribute and decision attribute to set up a policy table, then to delete the repeated object in the policy table and do condition property reduction of policy table;

3) Least reduction of condition property, obtaining the policy rules, then, appraisal and amalgamation of policy rules. The amount of policy rules from macro-policy to electric consumption is 39.

4) Fuzzy illation as a control problem: doing fuzzy illation based on policy rules in fuzzy toolbox, and to embed the feedback mechanisms into fuzzy illation method, renewing the rules according to the change of external environment.

5 Simulation and Experimental Analysis

We apply the DIC theory to policy simulation, through setting up an intelligent simulation platform based on agent to complete the simulation from policy to electric power supply and demand.

System input variable are some policy parameter, fuzzy language is described as constrictive, moderate or positive.

In monetary policy, lending rate is percentage and its value range is [5.31, 10.98]. Reserve rate is percentage and its value range is [6, 18]. Money supply (M2) is growth rate within the same period and its value range is [12, 20]. In financial policy, government expenditure is growth rate within the same period and its value range is [9.8, 24.8]. Revenue is growth rate within the same period and its value range is [8.9, 26.7]. Net exports is growth rate within the same period and its value range is [-1, 40].

System simulation results (output variable) include GDP and electric power consumption, fuzzy language is described as low, moderate or high. GDP is growth rate and its value range is [7.6, 14.2], electric power consumption is growth rate and its value range is [9, 13].

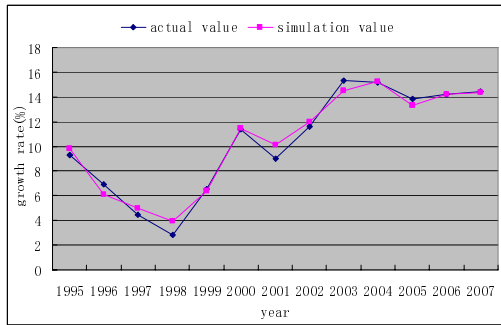


Fig. 5. Simulation results comparison

In order to verify the veracity and validity of intelligent simulation, policy infection to electric power consumption in 1995-2007 is simulated. Actual history macro-policy value from 1995 to 2007 are input into simulation system, simulation value of electric power consumption growth rate will be gained through the intelligent illation. Compare the simulation value with the actual value of electric power consumption growth rate, Fig. 5 shows that their trend is consistent and the agreement index is high.

6 Conclusion

The paper attempts to solve the complicated distributed system problems by DIC based on IE theory, and the experimental analysis verifies that this way is feasible and effectual.

1. Simulation results show that active financial policy and a moderate monetary policy can be effective in accelerating the investment, consumption and export growth, and the rapid industry development make the growth rate of electric power demand maintain a high level about 14.3%.
2. To transform the positive financial policy gradually to the moderate, and monetary policy will continue to be moderately tight, consequently, and the economic growth rate of secondary industry, heavy industries, steel industries, construction industries and nonferrous metals industries will be slowing down, and the growth rate of electric power demand will fall down.
3. The first half of 2008, the government of China put positive financial policy and tight monetary policy in practice, we set that lending rate is 7.47%, Reserve rate is 17.5%, Money supply (M2) growth rate is 17.37%, Government expenditure growth rate is 30%, Revenue growth rate is 30.5%, Net exports growth rate is 21.9%, simulation results show the electric power consumption growth rate will decrease to about 11.8% in 2008.

References

1. Song, J., Yu, J.Y.: On Stability Theory of Population Systems and Critical Fertility Rates. *Mathematical Modelling* 2, 109–121 (1981)
2. Luenberger, D.G.: A Nonlinear Economic Control Problem with A Linear Feedback Solution. *Automatic Control* 20(2), 184–191 (1975)
3. Zeng, Q.: Natrual Cybernetics. *Bulletin of the Chinese Academy of Sciences* 11(1), 16–21 (1996)
4. Fu, K.S.: Learning Control System and Intelligent Control System: An Intersection of Artificial and Automatic Control. *IEEE Trans. On AC* (February 1971)
5. Ishizuka, M., Kobayashi, S.: *Expert System*. Maruzen, Tokyo (1991)
6. Zadeh, L.: Fuzzy Sets. *Information and Control* 8, 338–353 (1965)
7. Hu, Z.: Studying on the Baseline Space of Sustainable Power Development. *Electric Power (Chinese)* 37(4), 1–4 (2004)
8. Hu, Z.: Intelligent Engineering-Its Application. In: *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, vol. 1, pp. 609–614 (1995)
9. Bin, Q., Xin, W., Min, W., Chun-hua, Y.: Framework of Distributed Integrated Control System Based on MAS and Its Prototype System Development. *Computer Integrated Manufacturing System (Chinese)* 12(10), 1633–1644 (2006)
10. Pawlak, Z.: Rough sets. *International Journal of Computer and Information Sciences* 11(15), 341–356 (1982)
11. Yu, X., Chen, G.: Discretization Behaviors of Equivalent Control Based Sliding Mode Control Systems. *IEEE Trans. Autom. Control* 48(9), 1641–1646 (2003)

Design of Directory Facilitator for Agent-Based Service Discovery in Ubiquitous Computing Environments

Geon-Ha Lee, Yoe-Jin Yoon, Seung-Hyun Lee, Kee-Hyun Choi,
and Dong-Ryeol Shin

School of Information and Communication Engineering,
Sungkyunkwan University,
300 Cheoncheon-dong, Jangan-gu, Suwon, Gyeonggi-do 440-746, Korea
{gh1ee, yoe21c, lshyun0, gyunee, drshin}@ece.skku.ac.kr

Abstract. As the ubiquitous computing is rapidly changing, the research on agent technologies is constantly being conducted. In a multi-agent system environment, each agent is registered in directory facilitator in a fixed form for service it provides, and gains a service function capable of modifying and deleting the service. Other agents may inquire about a service they want and receive necessary information on the service. By using this directory facilitator, the user can retrieve the most appropriate service. In this paper, we propose an efficient directory facilitator architecture that can improve the existing agent-based service discovery.

Keywords: directory facilitator, service discovery, CALM, multi-agent systems.

1 Introduction

Ubiquitous computing environment which is achieved to maximize human convenience has been developed despite the side effects such as infringement of privacy and security. In order to optimize the performance of ubiquitous computing and solve the mentioned problems, the development of the middleware that can provide services among heterogeneous hardware and operating systems is essential. Ubiquitous computing aims to provide integrated services through the context awareness of the users and devices forming a large-scale distributed system. The ubiquitous computing middleware operates independent from the hardware and software platform, and needs to efficiently provide reconfigurability, reusability, and adaptability in the level of operating system, middleware, and application in real time. This requires the middleware to be developed based on the component-based software. Such middleware provides an optimal service by collecting the context information and learning, inferring, and predicting the users' intention and preferences.

In this paper, we propose advanced directory facilitator based on CALM. The proposed system is a kind of directory service and can differentiate, register, manage, and store the agent service which was generated in agent platform, and when necessary it affords the base environment to search the target agent.

The remainder of the paper is organized as follows. Section 2 describes introduction of the Foundation for Intelligent Physical Agents (FIPA) and Java Agent

Development Framework (JADE). Section 3 presents our proposed mechanism and system architecture in detail. In Section 4, we implement the Directory Facilitator (DF) prototype. Section 5 presents the conclusion.

2 Related Work

In this section, we describe the background and related work. The FIPA is described in subsection 2.1, and the DF of the JADE agent platform as a FIPA-compliant agent system is described in subsection 2.2.

2.1 Foundation for Intelligent Physical Agents (FIPA)

FIPA is an IEEE computer standard organization with the aim of improving interoperability among heterogeneous agent - based technologies. The roles of FIPA are to establish the standards for agent technologies and standardize communications and management of agents [1].

Agent platform consists of Agent Management System (AMS), Directory Facilitator (DF), and Message Transport System (MTS). The main role of AMS is managing agent's creation, deletion and immigration on the agent platform. The AMS also maintains the index, which includes all agents AID (Agent Identifier) on the agent platform. The DF functions as a yellow page service on agent platform. The DF stores service descriptions offered from an agent, which operates as a service provider. The DF is an optional component of the agent platform. But, it is essential to easily find the service's location on the multi-agent system. Agents can their services with DF and register the service that enables an agent to search another agent providing a specific service on DF and inquire it [2]. The MTS is essential for communication between agents on different agent platforms.

2.2 Directory Facilitator

JADE is the middleware developed by the TILAB in order to develop Java - based distributed multi - agent applications based on the P2P communication architecture [3]. The communication between peers through these JADE platforms is done by the exchange of ACL (Agent Communication Language) between agents irrespective of whether it's wired or wireless network environments. JADE platform has containers to hold the agents, and a main container resides on the host which runs the RMI server of the platform. The agents are implemented as Java threads and live within the agent containers providing runtime support to the execution of the agents. The Main Container is referred to special container in the Platform. This container includes the AMS and DF.

The DF is a class of yellow page service. That is, the DF provides the agent's service repository. At first, agents create descriptions of their services and store these descriptions in the DF. The user can obtain the desired service's descriptions by retrieving the DF. The Knowledge Base in the DF is a real repository for storing service descriptions. The Knowledge Base stores DFAgentDescription objects that show the

descriptions of an agent. Similarly, a DFAgentDescription object can store ServiceDescription objects which are descriptions of the service provided by the agent. JADE’s existing DF uses concrete matching mechanism [4] for service discovery. This mechanism uses a syntactic and structural matching mechanism. That is, the sequential search mechanism is used. However, the DF’s Knowledge Base may contain too many DFAgentDescriptions, by different mobile agents in the ubiquitous environment. Therefore, it follows, that the DF should conduct all matching processes about all DFAgentDescriptions in the Knowledge Base. This generates problems in searching speed.

3 Proposed Architecture

In this section, we explain in detail the CALM-based DF, which is provide yellow page service in platform, efficient DF mechanism and system architecture.

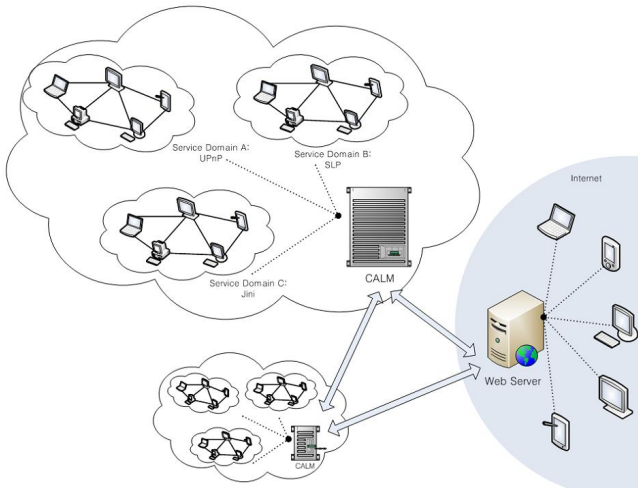


Fig. 1. The Conceptual Model

3.1 Efficient DF Mechanism

The proposed DF mechanism provides users with scalability of services, without any modification of existing service discovery technologies. Figure 1 depicts the conceptual model of the web-based integrated DF system. The architecture is composed of three environments. First, there are domains connected with service discovery protocols such as UPnP [7], SLP [8] and Jini [9]. Second, there is a CALM environment, as shown in figure 2, which detects messages of services existing within each service discovery protocol. Finally, there is a web server that supports the viewing of available services and invoking these services. The web server requests an available service list to the preregistered agent platform, according to the client’s request, and then presents the result as a web page.

3.2 System Architecture

The proposed DF architecture is used for a part of Component-based Autonomic Layered Middleware (CALM) project. Detailed information of CALM, is shown in [10]. Figure 2 shows our proposed DF architecture, illustrating the above concepts. The proposed architecture consists of Message Processor, Description Database, DF Function, Context Manager, Service Matching, Publish/Subscribe, and Web Interface module. The role of each component is as follows:

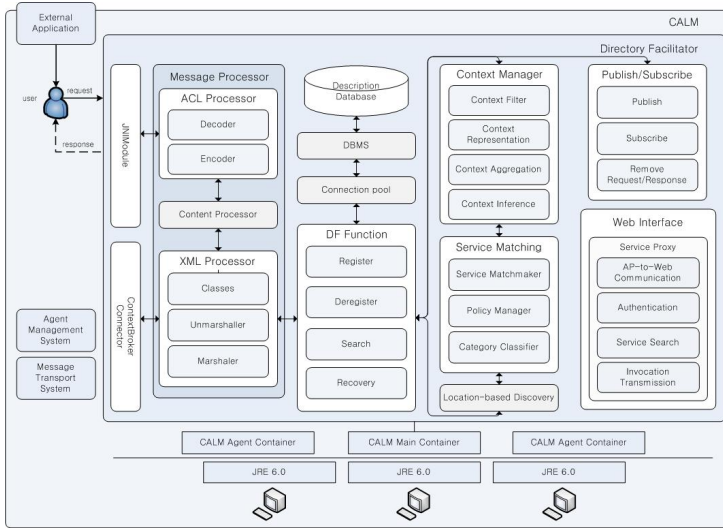


Fig. 2. Proposed DF architecture

* Message Processor

ACL Processor. The ACL Message Processor performs FIPA-ACL message parsing or composing for suitable CALM-based DF format to register the service description.

Content Processor. The Content Message includes a real command for operating the CALM-based DF. The content message is included in the FIPA-ACL message, and transmitted to the CALM-based DF. That is, the Content Message Parser performs the parsing task of the Content message.

XML Processor. The XML Processor performs ServiceDescription of FIPA-ACL message marshalling or unmarshalling for suitable CALM-based DF format to register the Description Database. Figure 3 shows process of the proposed converting module from FIPA SL to XML sequentially.

* Description Database

The Description Database has the role of storage or management the ServiceDescription received by service agents. This module performs task relevant to ServiceDescription as service repository through DBMS and Connection pool. Through DF Function module's operation, Description Database's ServiceDescription is managed.

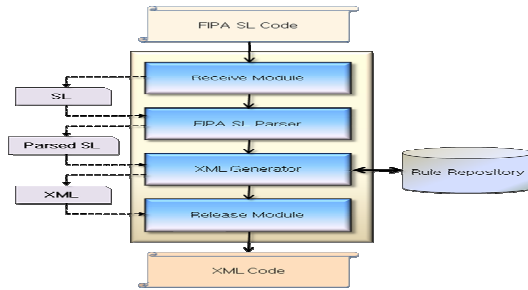


Fig. 3. Converting module from FIPA-SL to XML

*** DF Function**

The DF Function Module performs task relevant to service description as service repository of the FIPA-compliant agent system. These tasks include service description’s registration, deregistration, modification and search function. Through this module, service description is stored in the Description Database of the CALM-based DF.

*** Context Manager**

The Context Manager has the role of filtering or inferencing the different contexts received by various agents. The Context Manager consists of Context Filter, Context Inference Module and Context Repository. The Context Filter’s primary purpose is to protect the DF from being flooded with excessive information. The Context Inference Module creates a new context from existing raw contexts. The Context Inference Module is needed because not all information can be extracted from raw context. It contains diverse rules to create new context.

*** Service Matching Module**

Service Matchmaker. The results provided by the Context Manager are forwarded to the Service Matchmaker in the Service Matching Module. The Service Matching Module consists of Service Matchmaker, Policy Manager, Category Classifier and two databases.

Category Classifier. From the Category Database, Category Classifier can create the available addresses which are stored or searched in the Description Database. Category Classifier’s roles are twofold as both registration and search. When ServiceDescriptions are categorized, search time is the sum of the Category Database’s seek time and ServiceDescription. That is, the search time is lower than the existing sequential mechanism, according to the increase in the number of agents.

Policy Manager. The Policy Manager stores pre-defined service policies offered by service provider in the Policy Database. The services registered in the Description Database are managed by the Policy Manager using the policies stored in the Policy Database.

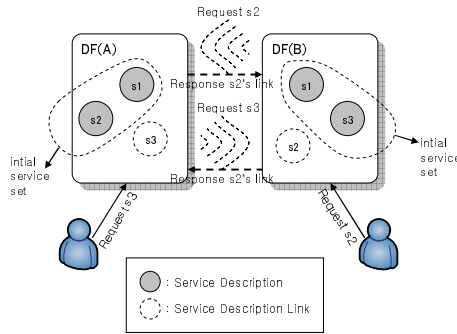


Fig. 4. Publish/subscribe scenario

*** Publish/Subscribe**

The Publish/Subscribe module has the role of publishing or subscribing the agent-based service received by it for dynamic service discovery. The Publish/Subscribe module performs task applying to the integrated storage in different agent platforms, not to service storage in local. An advantage of this model is that user can utilize and subscribe the services not only in local network but also in different network. Figure 4 is scenario of Publish/Subscribe model among request agents.

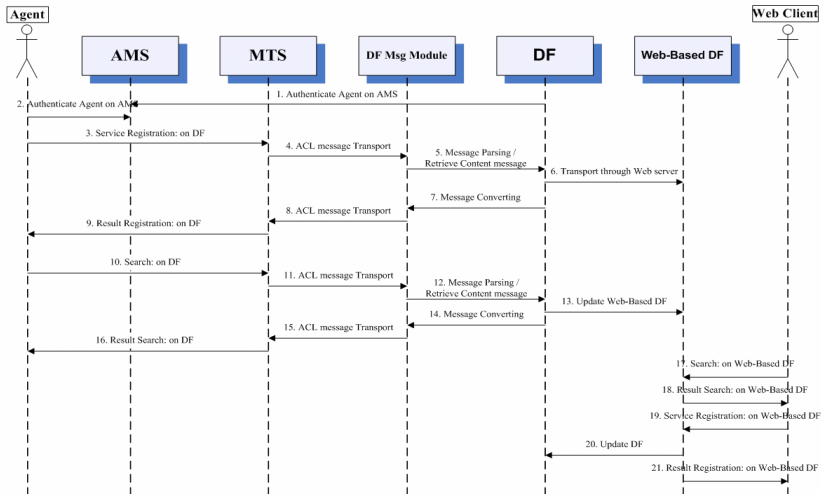


Fig. 5. Sequence diagram of Web Interface

*** Web Interface**

The purpose of Web Interface module is to differentiate, register, and store the agent service that was generated in agent platform by connecting web browsers which are widely distributed in general and has the principle to share information between web server and DF. We illustrate the sequence diagram in figure 5 especially in case of discovering the services located in agents.

4 Implementation

Implementation of the proposed DF prototype is based on Java. Figure 6 indicates the web page for searching the ServiceDescription and the DF result page of a user search request.

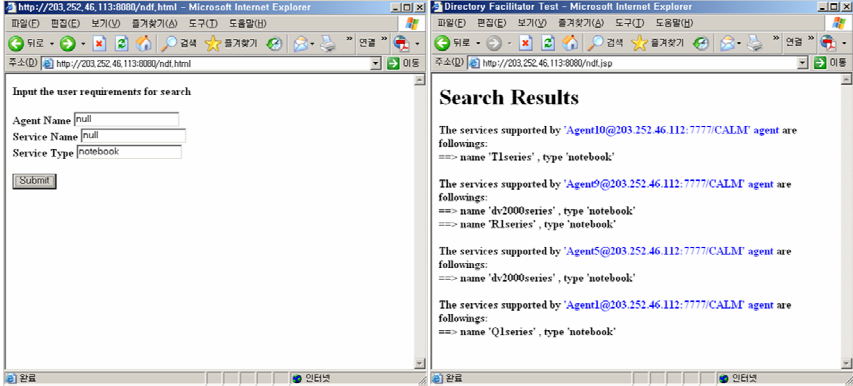


Fig. 6. DF search and result page

5 Conclusion

This paper mainly contributes to present an advanced DF architecture for agent-based service discovery in heterogeneous environments. We analyzed the JADE's DF which is the repository of service descriptions. Similarly, we pointed out the problems of the existing DF mechanism for storing and searching the services. After consideration of these problems, an advanced DF architecture for agent-based service discovery is proposed. The proposed architecture is more efficient than existing mechanisms, when increasing the number of agents in multi-agent system. In future work, we will compare the proposed architecture with other searching mechanisms and evaluate the performance.

Acknowledgements

This research is supported by Foundation of ubiquitous computing and networking project (UCN) Project, the Ministry of Knowledge Economy(MKE) 21st Century Frontier R&D Program in Korea and a result of subproject UCN 08B3-B1-10M.

References

1. FIPA: The Foundation for Intelligent Physical Agents, <http://www.fipa.org>
2. Caire, G.: JADE Tutorial (2003), <http://jade.tilab.com>

3. Bellifemine, F., Bergenti, F., Caire, G., Poggi, A.: JADE-A Java Agent Development Framework, Multi-agent Programming. In: Bordini, R.H., Dastani, M., Dix, J., El Fallah-Seghrouchni, A. (eds.) *Multi-Agent Programming: Languages, Platforms and Applications, Multi-Agent Programming. Multiagent Systems, Artificial Societies, and Simulated Organizations*, vol. 15, pp. 125–147. Springer, Heidelberg (2005)
4. Naumenko, A., Nikitin, S., Terziyan, V.Y.: Service matching in agent systems. *Appl. Intell.* 25(2), 223–237 (2006)
5. Efstratiou, C., Cheverst, K., Davies, N., Friday, A.: An architecture for the Effective Support of Adaptive Context-Aware Applications. In: *Mobile Data Management*, pp. 15–26 (2001)
6. Lee, K.M., Kim, H.-J., Shin, H.-J., Shin, D.-R.: Design and Implementation of Middleware for Context-Aware Service Discovery in Ubiquitous Computing Environments. In: Gavriloa, M.L., Gervasi, O., Kumar, V., Tan, C.J.K., Taniar, D., Laganá, A., Mun, Y., Choo, H. (eds.) *ICCSA 2006. LNCS*, vol. 3983, pp. 483–490. Springer, Heidelberg (2006)
7. Universal Plug and Play Specification, v1.0, <http://www.upnp.org>
8. Guttman, E.: Service Location Protocol – automatic discovery of IP network services. *Internet Computig Journal* 3(4) (1999)
9. Jini Architecture Specification, v1.2, <http://www.sun.com/software/jini/specs/>
10. Han, S., Song, S.K., Youn, H.Y.: CALM: An Intelligent Agent-based Middleware for Community Computing. In: *Proceedings of the Fourth IEEE Workshop of SEUS 2006 /WCCIA 2006* (2006)
11. Lee, K.M., Kim, D.-U., Choi, K.-H., Shin, D.-R.: Web-based Integrated Service Discovery Using Agent Platform for Pervasive Computing Environments. In: Shi, Y., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) *ICCS 2007. LNCS*, vol. 4488, pp. 952–955. Springer, Heidelberg (2007)
12. Kim, D.-U., Heo, S.-P., Lee, G.-H., Choi, K.-H., Shin, D.-R.: Design of Agent-based Integrated u-Healthcare System supporting Heterogeneous Services. In: Torra, V., Narukawa, Y., Yoshida, Y. (eds.) *MDAI 2007. LNCS (LNAI)*, vol. 4617, pp. 50–59. Springer, Heidelberg (2007)
13. Kim, D.-U., Lee, G.-H., Lee, K.M., Heo, S.-P., Choi, K.-H., Shin, D.-R.: Design and Implementation of Efficient Directory Facilitator for Context-Aware Service Discovery. In: Nguyen, N.T., Grzech, A., Howlett, R.J., Jain, L.C. (eds.) *KES-AMSTA 2007. LNCS (LNAI)*, vol. 4496, pp. 588–596. Springer, Heidelberg (2007)
14. Lee, S.-H., Jang, K.-S., Shin, H.-J., Shin, D.-R.: Agent-Based Discovery Middleware Supporting Interoperability in Ubiquitous Environments. In: Nguyen, N.T., Grzech, A., Howlett, R.J., Jain, L.C. (eds.) *KES-AMSTA 2007. LNCS (LNAI)*, vol. 4496, pp. 141–149. Springer, Heidelberg (2007)

Laboratory of Policy Study on Electricity Demand Forecasting by Intelligent Engineering

Zhaoguang Hu, Minjie Xu, Baoguo Shan, and Xiandong Tan

State Power Economic Research Institute of China
No.1 Ertiao, Baiguang Road, Xuanwu, Beijing, 100761 China
{huzhaoguang,xuminjie}@chinasperi.sgcc.com.cn

Abstract. Electricity demand will be affected by national policies and other factors. There are many semi-structure problems in the electricity demand forecasting, which are very difficult to be solved by the use of traditional methods. In this paper, intelligent engineering is developed. It adopts theory and technique of artificial intelligent, soft computing, uncertain theory, and multi-agent system. Three fundamental problems and generalized model are proposed in intelligent space. As a case, inspired by the physical experiment, the laboratory of policy study is built based on intelligent engineering to simulate the impact of the national policy on electricity demand forecasting. A case study in China has been shown in the paper.

Keywords: Intelligent engineering, Intelligent space, Generalized model, Laboratory of policy study, Electricity demand forecasting.

1 Introduction

With two digits growth of Gross Demand Products (GDP) in the last five years in China, the growth rate of electricity demand was higher than that of GDP. The power shortage had been happened during the years 2003-2006. So, the electricity demand forecasting is important to balance power supply and demand. On the other hand, Chinese government tried to control the fast growth of GDP by monetary and financial policies. There are lots factors which will affect electricity demand such as national economic growth, international trade and national policies. The relation between electricity demand and policies such as monetary policy, financial policy, and international trade are quite complex. It used to be studied by computable general equilibrium (CGE) model [1] which is a non-leaner mathematical model. However, since CGE model is based on Input/Output table, the table is reviewed every five years in China. The latest table in China is the version of 2002. It is clear that the economic structure and state have been varied greatly since 2002 in China. It is the challenge for CGE model.

Fortunately, Artificial Intelligence (AI), Neural Networks (NN), Fuzzy Systems (FS) and other advanced technologies have been studied to provide the way to model these complex systems. However, the techniques are performing with advantages and limitations respectively. Intelligent engineering [2][3] combine all of the techniques together to share the advantages and avoid the limitations in modeling complex

systems. In the paper, IE is extended, and three basic problems and general models are proposed in intelligent space. The policy test can be simulated in the laboratory for electricity demand forecasting.

2 Intelligent Engineering

2.1 Intelligent Engineering Theories

The fundamental of intelligent engineering (IE) in theory has been studied in literature [2, 3] as follows.

Definition 1. Intelligent path P is a set of (fuzzy) relations, (fuzzy) mappings, transformations and all the ways between the start state set S_0 and destination state set S_n . It can be in the form as $P : S_0 \rightarrow S_n$.

Definition 2. Let $d \in S_n$ and $s \in S_0$, P is a set of intelligent path between S_0 and S_n . Then, intelligent equation is in the form as $d = IP(s)$.

Definition 3. Intelligent space is defined as $I = (P, S)$.

Where S is the set of state, and P is the set of intelligent path between S_0 and S_n , S_0 and S_n are subset of S .

Definition 4. A problem B is defined as $B = (S_0, S_n, PB)$.

where S_0 is a set of start states, S_n is a set of destination states and PB is a set of the intelligent path between S_0 and S_n , it is a subset of P in intelligent space I . $PB \subset P$.

Definition 5. A α -smart solution for the problem B is defined as $SL(\alpha)$, if there is a fuzzy set fp in PB ,

$$fp : PB \rightarrow [0,1] \text{ and } SL(\alpha) = \{x \mid \mu_{fp}(x) \geq \alpha, x \in PB\}.$$

where $\alpha \in [0,1]$. There are three types of problem B in intelligent space: ① B1: from known initial state and known intelligent path to forecast target state $(S_0, PB) \rightarrow S_n$,

② B2: from known initial state and known target state to search intelligent path $(S_0, S_n) \rightarrow PB$,

③ B3: from known target state and known intelligent path to verify initial state $(PB, S_n) \rightarrow S_0$.

The forecasting, scenario analysis and planning can be modeled as problem B1. The strategy problem can be modeled as B2. The reviewing history state can be modeled as B3.

2.2 Generalized Model

Here are three different kinds of the model in the IE as follows:

(A) AI rule based model: The rules to express the knowledge form as “IF A THEN B”. It is the form same as expert systems. It can be expressed as mapping: $f : X \rightarrow Y$, where A is an element in X and B is an element in Y. Knowledge engineers acquire, store and process the bivalent rules as symbols, but not as numerical

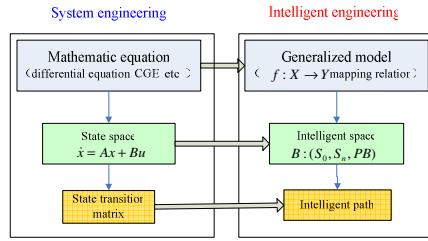


Fig. 1. Comparison of IE and System engineering

entities. This often allows knowledge engineers rapidly acquire structured knowledge from experts and to efficiently process it. But it forces experts to articulate the proposition rules that approximate their expert's behavior.

(B) NN model: The form to express the knowledge in the NN model. It can also be expressed as mapping: $f : X \rightarrow Y$, where X is a set of inputs of NN model and Y is a set of outputs of the model. A trained NN model can express the human's knowledge that is difficult or cannot be expressed in the AI rule form. This kind of knowledge can be acquired from history data or samples. Then, the NN model can be trained by the data with/without human experts' supervision. In this way, it is very easy for knowledge engineers to acquire the knowledge, and for intelligent system to learn the new knowledge automatically by training the NN model with new data.

(C) Fuzzy model: The form to express the knowledge with uncertainties, such as fuzzy rules, fuzzy neural networks. As we know that there is lots of uncertainty knowledge and experiences that even human experts cannot express them clearly. They can be in the form as if A then B , where A and B are fuzzy sets in X and Y respectively.

Mathematical model and above three kinds of model can be called as generalized model. IE is development of system engineering, and not only includes analysis methods of system engineering, but also combines artificial intelligence, soft computer, uncertain theory, and multi-agent system, etc. In IE theory, mathematic model is extended to generalized model, state-space is extended to intelligent space, and solving of state transition matrix is replaced by intelligent path solving (Fig. 1). So, IE is a methodology for complex system with generalized model.

3 The Structure of Laboratory of Police Study

3.1 Principles of the Laboratory

On policy study, can the problems be experimented as physical experiment by computer? With development of computer science and innovation of methodology, the ideal is realized by methodology of intelligent engineering.

The laboratory of policy study on electricity demand forecasting is soft-science laboratory based on IE. The laboratory consists of knowledge base, intelligent inference, methodology base, model base, data base, and interface, as shown in the Fig. 2.

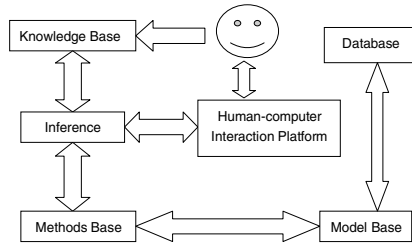


Fig. 2. The elementary diagram of intelligent laboratory

The database is a complex data entity of management which is object oriented. It is used for the storage of the data information needed by simulation test system, and is the data source needed by the operation of the simulation model. The model base is used for the management and storage of various simulation models, involving forecasting model, optimization model, reasoning model, and so on. All the models mentioned above compose the simulation model system of the intelligent simulation laboratory, and is for the method base call. The methods base is used for the storage and management of the methods which are adopted to support users' optimal decision-making. The intelligent inference is the most key component in intelligent laboratory, which is connected with the knowledge base, and has reasoning function. It can realize the mapping relation of generalized model and the user can design simulation scheme reasonably by using the Human-computer interaction platform.

The knowledge base is used for the storage of the knowledge rules, which describes the map of primates and consequences and is denoted the generalized model, i.e. $f : X \rightarrow Y$.

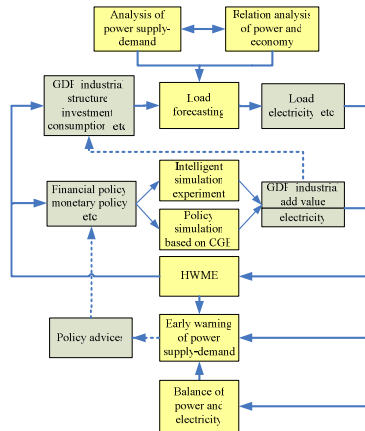


Fig. 3. Relationships of module

3.2 Structure of the Laboratory

The functions of the laboratory are analysis of the main factors of influencing the macroeconomic development and the influence of the production and transportation of

energy, weather and water on power supply and demand by the quantitative and qualitative methods based on the data acquisition of power, energy, economy, weather, and water. It combines the experience and wisdom of the experts. As a whole, the functions are consist of analysis of power supply-demand, relation analysis of power and economy, load forecasting, balance of power and electricity, early warning of power supply-demand, policy simulation, intelligent simulation experiment, and HWME (hall for workshop of meta-synthetic engineering). Relationships of function module are shown in Fig. 3.

3.3 Acquisition of Knowledge

The acquisition and expression of knowledge are the key of the laboratory, and the core of simulation inference. The method study of the knowledge acquisition and expression is the basic of Artificial Intelligence.

3.3.1 Rule Mining Based on Rough Set

The rough set theory proposed by [4] provides an effective tool for extracting knowledge from data tables. The rough set theory extends classical set theory, embeds knowledge used for sorting in the set, and accordingly makes it become a part of set. Knowledge mining by using rough set doesn't need priori-knowledge, and is able to solve and express incomplete even conflicting information, finally could find a set of concise rules by knowledge reduction. This method is widely applied in engineering.

3.3.2 Generating Rules System Based on Feedback

Knowledge reduction based on rough set could find out some rules which can describe development of the things from historical data, and forms the base based on rule system.

For the dynamic external world, history summary could provide future decision with important guide, but when external environment changes, robustness of inference system will become the key of advisement. Moreover, fuzzy rule base reduction based rough set may be make undesirable errors in the mapping from fuzzy premises to fuzzy consequences. Robustness of fuzzy reasoning is concerned with the effects of perturbations associated with given fuzzy rule bases and/or fuzzy premises on fuzzy consequences. In order to improve robustness of fuzzy reasoning, fuzzy reasoning were treated as a control problem and feedback mechanisms were embed into fuzzy reasoning method. The reasoning method involves an explicit feedback mechanism and corresponds to a closed-loop reasoning system [5] is depicted in Fig. 4. More specifically, the given fuzzy rule base serves as the controlled object, and the fuzzy reasoning method serve as the corresponding controller. The fuzzy rule base and the reasoning method constitute a control system, which may be open-loop or closed-loop, to achieve or satisfy given reasoning goal and constraints. The fuzzy rule base, the reasoning methods, and the reasoning goals and constraints specify the three distinct ingredients of fuzzy reasoning. By the inference optimization idea based on feedback, it could be adjusted to make external environment better according to evaluation dynamic renewed rule set of the object. It carries through optimizing rule by using fuzzy inference as a control problem.

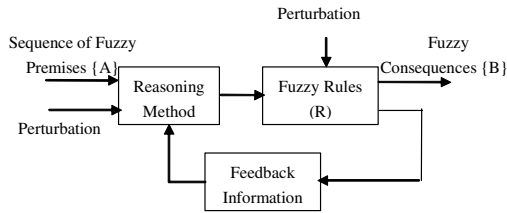


Fig. 4. Diagram of Closed-loop Reasoning

3.3.3 Collecting and Summing Up Expert's Knowledge and Experience

The main method of knowledge acquisition is data mining, namely finding out well regulated knowledge from abundant historical data by using data mining algorithm. In China, the statistical datum is not full and statistical calibers are not uniform in the early of 1980s. So seeking the relation between national policy and economic state, electric power consumption only with historical sequence can not get perfect result. Since socioeconomic system is a huge complex system which is made of various coupling factors, the promulgation and implementation of various policies, the effect of the policy implementation can not be well reflected by statistic data. So collecting and summing up domain expert's knowledge and experience would become important complementary instrument. In 1992 the academicians Xuesen Qian brought forward the idea "HWMSE" [6], which combined expert system, data and information system, and computer system organically to be a meta-synthetic environment on a certain question and got rules from that not only to make judgment on certain a question but also to extend knowledge base to be decision rule of general user's test.

4 Experimental Case Studies

As a case, influence of policy on electricity demand is simulated in the laboratory. First, the inference rules were found out based on historical data by rough set, generating rules system, and HWME. Second, the change of electricity demand was analyzed under the action of various policy sequences which can be expressed as intelligent paths. In intelligent space, the process of electricity demand forecasting can be expressed as problem *B1*.

4.1 Experimental Design

The experiment studies the change of electricity demand in the coming year under the action of various national policy sequences as follows: ① Monetary policy: monetary stringency, monetary non-stringency and non-ease, monetary ease. ② Financial policy: fiscal restraint, fiscal balance, fiscal expansion. ③ International trade: high growth, growth, lower growth.

We can see that monetary policy will affect economic activity (production). For example, monetary ease policy will promote manufacture's production, and then drive to use more electricity. So, monetary policy will affect the electricity demand. We can do the same analysis for fiscal policy and international trade.

It shows the effect which different sequences make on the growth rate of electricity demand. In intelligent space, it expresses as initial state S_0 and scenarios of possible policy (intelligent path PB), and simulation of the states of $S_{n1}, S_{n2}, \dots, S_{nm}$ (Fig.5). For the conduction process from national policy to power demand is abstract and complicated. We synthetically use the methods of knowledge mining and extracting rules based on the policy sequence and electricity consumption growth sequence from 1990 to 2007. First of all, initial rule set R_1 was got by the rough set software—ROSETTA [7]. Secondly, the R_1 was corrected to R_2 that it was considered as a feedback control problem [5]. Finally, adopting and absorbing expert’s knowledge and experience, the R_2 was perfected by HWME [6].

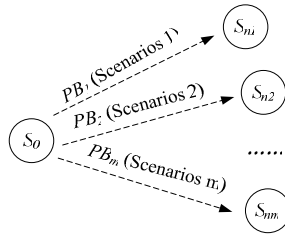


Fig. 5. Intelligent paths of simulation

4.2 Scenarios Design

There will be some different policy sequences which can be designed as scenarios as follows:

Scenarios 1: It is possible for the officer to select monetary policy in the scale near the monetary stringency. Achievements of macro-control are notable in 2007, in order to consolidate these achievements, financial policies with fiscal balance will be selected. Assuming that international economic situation remains to be in neutral position, it is possible that growth rate of international trade in China is higher than that in last year.

Scenarios 2: Comparing with Scenarios 1, monetary policies with monetary non-stringency and non-ease and financial policies with fiscal balance are selected, and growth rate of international trade is near the “growth”.

Scenarios 3: Comparing with Scenarios 1, monetary policies with monetary stringency and financial policies with fiscal restraint are selected, and growth rate of international trade is lower than that in last year.

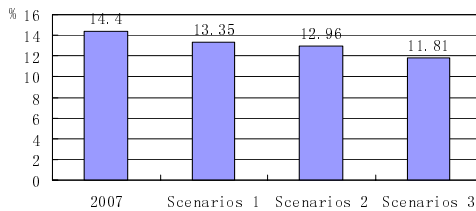


Fig. 6. Simulation results

The results of electricity demand are shown in Fig. 6 based on growth of electricity demand in 2007. In Scenarios 1, the growth rate of electricity demand will be about 13.35% in the 2008. In Scenarios 2 and 3, the growth rate of electricity demand will be about 12.96%, 11.81%, respectively. It is shown that the both monetary policy and fiscal policy play an important rule in the cooling of hot economy.

5 Conclusions

The electricity demand forecasting is a quite difficult issue to policy study. Even it is impossible to get very good results since there are lots factors and uncertainties. Intelligent engineering is a methodology of complex problems which difficult to be modeled in mathematics. IE shares the theory and technique of artificial intelligent, soft computing, uncertain theory, and multi-agent system. In this paper, three fundamental problems and generalized model are studied in intelligent space. The laboratory of policy study on electricity demand forecasting is built based on intelligent engineering theory to model the relation between policies and electricity demands because there are many semi-structure problems which are very difficult to solve by the use of traditional methods. In the laboratory, users can focus on construction of generalized model for specific simulation task, and simply set up parameter for scenarios simulation. A few scenarios were simulated for analyzing electricity demand in China. Simulation results show that the laboratory is a useful tool for officers to test the effect of national policies.

References

1. Zheng, Y., Fan, M.: CGE Model and Policy Analysis in China. China Social Sciences Press, Beijing (1999)
2. Hu, Z.: Studying on the baseline space of sustainable power development. *Electric Power (Chinese)* 37(4), 1–4 (2004)
3. Hu, Z.: Intelligent engineering-its application. In: *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, vol. 1, pp. 609–614 (1995)
4. Pawlak, Z.: Rough sets. *International Journal of Computer and Information Sciences* 11(15), 341–356 (1982)
5. Cai, K., Zhang, L.: Fuzzy reasoning as a control problem. *IEEE Trans. on Fuzzy System* (in press)
6. Yu, J.: Qian Xuesen's contemporary system of science and technology and meta-synthesis. *Engineering science* 3(11), 10–18 (2001)
7. Øhrn, A.: *Discernibility and RoughSets in Medicine: Tools and Applications*. PHD. thesis. Trondheim, Norway

Self-adaptive Mutation Only Genetic Algorithm: An Application on the Optimization of Airport Capacity Utilization

King Loong Shiu and K.Y. Szeto

Department of Physics,
Hong Kong University of Science and Technology,
Clear Water Bay, Hong Kong SAR, China
phszeto@ust.hk

Abstract. A new type of adaptive evolutionary algorithm that combines two genetic algorithms using mutation matrix is developed based on an adaptive resource allocation of CPU time. Performance evaluations are made on the airport scheduling problem with constraint. The two genetic algorithms used are based on the construction of the mutation matrix $M(t)$, which is problem independent as it uses the fitness distribution in the population and the statistical information of the locus only. The mutation matrix is parameter free and adaptive since the matrix elements are time dependent and inherits the information accumulated from past generations. A self-adaptive time sharing method is introduced to allocate resource to the two different strategies, which uses the theory of mean-variance analysis in portfolio management. The application to airport scheduling demonstrates that the self-adaptive mutation only genetic algorithm is able to provide quality solutions efficiently.

1 Introduction

Genetic algorithms and evolutionary computation [1,2] based on the Darwinian principle of “survival of the fittest” have been implemented successfully in many fields such as global optimization problems [3], cryptography [4] and time-tabling [5] with very promising results. However, the use of genetic algorithms usually requires intelligent choices of parameters, such as the survival selection ratio and the mutation probability. In most cases, the efficiency of the algorithm is closely related to the parameters chosen by the user [3], which depends on the user’s experience on the problem. The choice of the parameters may be good initially, but will usually become less efficient after some time in the evolution process. Indeed, the parameters should be time dependent for higher efficiency [6]. Based on the recent work on genetic algorithm using mutation matrix [6] and the extended version that includes crossover [7], we incorporate the idea of temporal resource allocation using the method of mean variance analysis in portfolio management in this paper.

In the first paper on mutation only genetic algorithm (MOGA) by Szeto and Zhang[6], they introduced two versions of evolution process using mutation matrix: MOGA *by Row* (MOGAR) or MOGA *by Column* (MOGAC). A quasi-parallel method to combine the above two algorithms into one that strikes a balance between

exploration (as emphasized by MOGAR) and exploitation (as emphasized by MOGAC) was designed so that the combined method is more preferable than any one of the component algorithms individually. The optimum value of the time sharing ratio between these component algorithms can be determined by a search on the “investment frontier” in a mean-variance analysis [8]. They call this mixing of MOGAR and MOGAC, MOGAM, which stands for Mutation Only Genetic Algorithm with Mixing. However, MOGAM still requires the user to determine the time sharing ratio. In this paper, we further develop MOGAM by eliminating the time sharing parameter. We use the accumulated statistics of the performance of MOGAR and MOGAC to determine the time sharing ratio that enables a low risk but high speed in the drift towards the solution. We test this parameter free approach on the optimization of airport capacity utilization problem as discussed by Gilbo [9] with satisfactory results with high efficiency. We first review the work on mutation matrix in section 2.

2 Mutation Only Genetic Algorithms

Consider a population of N chromosomes, each of length L to form a $N \times L$ matrix $A(t)$ for the population at a given generation t . We reorder the chromosomes in the matrix such that the i^{th} row represents the chromosome with fitness f_i , and $f_i \geq f_k$ if $i \leq k$. Then $A_{ij}(t)$, $i = 1, \dots, N$; $j = 1, \dots, L$ represents the value of the j^{th} locus of the i^{th} chromosome. Next, we introduce a mutation matrix $M(t)$ with elements $M_{ij}(t) = a_i(t)b_j(t)$, $i = 1, \dots, N$; $k = 1, \dots, L$, where $0 \leq a_i(t)$, $b_j(t) \leq 1$ are the row mutation probability and column mutation probability respectively.

2.1 Row and Column Mutation Probability

We first consider the case of a fit chromosome. We expect to mutate a few loci only so that it keeps most of the information unaltered. This corresponds to “exploitation” of the features of the fit chromosome. For the case of an unfit chromosome, we expect to mutate many of its loci in order to let it explore other regions of the solution space. This corresponds to “exploration”. Therefore, we require the mutation matrix $M_{ij}(t)$ to be a monotonically increasing function of the row index i as the chromosomes are arranged in descending order of fitness. For simplicity, we set $a_i = (i - 1) / (N - 1)$.

Next we must determine which loci to undergo mutation. We define p_{jX} as the locus mutation probability of the j^{th} locus changing to hold the value X . The value of p_{jX} is determined by the following equation:

$$p_{jX} = \frac{1}{\sum_{i=1}^N i} \sum_{i=1}^N (N+1-i) \times \delta_{ij}(X) \quad (1)$$

Here i is the rank of the chromosome and $\delta_{ij}(X)$ if $A_{ij}(t) = X$, and zero otherwise. The factor in the denominator is for normalization. We can see that p_{jX} contains information of both the row and locus statistics. Note also that the statistics is biased in such a way that a heavier weight is given to chromosomes with higher fitness. After obtaining p_{jX} , we define the column mutation rate as:

$$b_j = 1 - \frac{1}{C} \sum_x \left| p_{jx} - \frac{1}{D} \right| ; \quad C = 2 \left(1 - \frac{1}{D} \right) \tag{2}$$

Here D is the number of values X can take on (D is finite). The normalization factor is C to ensure that $0 \leq b_j \leq 1$. To interpret b_j , we first consider the case when X is follow a uniform distribution. In this case, $p_{jx_1} = \dots = p_{jx_D} = 1/D$ and $b_j = 1$. We interpret that we have no information about this j^{th} locus, thus we should mutate it. On the other hand, if $A_{ij}(t)$ assumes the value X_k for all j , then we have definitive information on the locus j and so we must not mutate it. In this case, it can be easily verified that $p_{jx_i} = 0$ if $i \neq k$ and $p_{jx_i} = 1$ if $i = k$. In this case, $b_j = 0$, and no mutation will be performed on this locus. Thus, b_j contains the statistical information on the j^{th} locus.

2.2 Evolutionary Strategies Using the Mutation Matrix $M(t)$

Once we have obtained a_i and b_j , the mutation matrix $M(t)$ can be immediately constructed by $M_{ij} = a_i b_j$. Now there are two ways to apply $M(t)$ on the population $A(t)$ for mutation. (a) We can first decide which row (chromosome) to be mutated and then which column (locus) to mutate. We call this method the Mutation Only Genetic Algorithm *by Row*, or MOGAR. (b) We may also first decide which column to be mutated, then which row to mutate. We call this the Mutation Only Genetic Algorithm *by Column*, or MOGAC. For MOGAR we perform mutation on the $N \times L$ population matrix $A(t)$ based on the following procedures:

1. For every row i , we generate a random variable z distributed uniformly on $[0,1]$
2. If $z < a_i(t)$, we perform mutation on this row; else, proceed to row $(i+1)$
3. Arrange $b_j(t)$ in descending order and choose the first $P_i(t) = a_i(t) \times L$ members
4. Perform mutation on these selected loci on the i^{th} chromosome to obtain $A_{ij}(t+1)$
5. Goto step 1 for row $(i+1)$

After we have gone through all N rows, we obtained $A(t+1)$. We then compute $M(t+1) = a_i(t+1) b_j(t+1)$ and proceed to the next generation of population $A(t+1)$. For MOGAC, the operation is similar to MOGAR except now we take the transpose of the matrix $A(t)$.

3 Self-adaptive Quasi-parallel Algorithm Based on the Idea of Investment Frontier

Although MOGAR and MOGAC have similar operational procedures, they actually differ in performance. MOGAR works better in exploring new regions of the solution space, at the price of slow convergence. MOGAC works better in exploiting fit chromosomes, at the price of early convergence. In order to maintain a balance between exploration and exploitation, Szeto and Zhang [6] have proposed the Mutation Only Genetic Algorithm with Mixing (or MOGAM), which is a ‘‘Quasi-Parallel Genetic Algorithm’’ to mix the two strategies to become one that outperforms both MOGAR and MOGAC when run separately. The quasi-parallel algorithm assumes that only a

single CPU resource is available. MOGAM uses a time sharing parameter $\gamma \in [0, 1]$ which is the fraction of time which the CPU executes MOGAR, with the remaining $(1-\gamma)$ fraction of time executing MOGAC. Thus, at a given generation, we generate a random number z to determine which algorithm is to be executed: if $z < \gamma$, use MOGAR; else, use MOGAC.

Although the quasi-parallel algorithm of MOGAM outperforms MOGAR or MOGAC in general, the requirement of setting the time sharing parameter γ is not satisfactory. In a brute force approach, one can perform MOGAM many times using different values of γ for a specific class of problem and then document the best parameter for future applications [6]. However, a much more satisfactory approach will be to avoid the presetting of the time sharing parameter with the help of adaptive parameter control. Our adaptive approach makes use of the idea of “investment frontier” in the theory of portfolio management in economics to locate the optimal time sharing ratio among two algorithms. Suppose we have two genetic algorithms G_1 and G_2 . We set $\gamma = 0.5$ as we have no information on which one is better. We then allow each of them to run for some generations to collect sufficient statistics, record the average performance (μ_1, μ_2) of G_1, G_2 and the associated variance of performance (σ_1^2, σ_2^2) over these generations,

$$\mu_n = \frac{1}{T_n} \sum_{m=1}^{T_n} \left(\sum_{i=1}^N (N-i+1) \Delta f_{i,m} \right) \tag{3}$$

$$\sigma_n^2 = \frac{1}{T_n} \sum_{m=1}^{T_n} \left\{ \left(\sum_{i=1}^N (N-i+1) \Delta f_{i,m} \right) - \mu_n \right\}^2 \quad ; \quad n=1,2 \tag{4}$$

Here $\Delta f_{i,m}$ is the increase in fitness of the i^{th} chromosome when the m^{th} time G_n is performed, and T_n is the total no. of times which G_n is performed. The factor $(N-i+1)$ is a weighting factor biased in favor of fitter chromosomes. We define $S_n = \mu_n / \sigma_n$ as the performance-volatility ratio of the algorithm G_n . We then define an adaptive time sharing parameter $\gamma^* = S_1 / (S_1 + S_2)$ where γ^* and $(1 - \gamma^*)$ are the fractions of time to execute G_1 and G_2 . This choice of γ^* aims at maximizing the returns over risk. Note that γ^* is a time dependent parameter since it depends on the S_n . The idea is to achieve high efficiency with small risk through temporal resource allocation on the single CPU. In our present case, MOGAR is G_1 and MOGAC is G_2 . We call this method the Self Adaptive Mutation Only Genetic Algorithm, or SAMOGA.

If compared with the two-armed bandit method [10], our adaptive parameter control takes into account the variance of the algorithm performance in our decision process on optimal time sharing. We observe that the performance of each component algorithm varies over a large range in the evolution process, convincing us that an intelligent and online method for good time sharing between component algorithm must take into consideration of the performance-volatility ratio. Notice that the value of the adaptive time sharing parameter γ^* contains the statistical information of past generations and itself is time and problem dependent. The advantage of our present method of time allocation is that we do not need to preset the parameter γ^* .

4 Optimization of Airport Capacity Utilization

We apply our method to the problem of optimizing airport capacity utilization [9]. In this problem, the arrival and departure capacity of an airport are dependent on each other. Their relationship is specified by an airport capacity curve, which plots the departure capacity against the arrival capacity. Here we use the Beijing International airport capacity curve [11] shown in Fig.1.

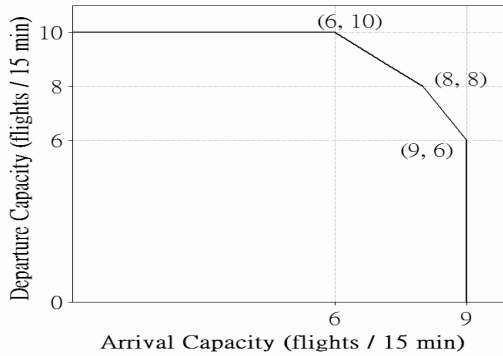


Fig. 1. Aiport Capacity Curve of The Beijing International Airport

Although flight delays are costly, we still have to minimize the delay cost while providing a safe schedule. Thus, any optimization scheme for the airport capacity utilization must not violate the airport’s schedule regulation to ensure safety. We also conform to the regulation which gives priority to flight arrival over flight departure [11], in order that the arriving planes can land as soon as possible. Usually, air delays are more costly and more dangerous. For flights scheduled at the same time slot, there is no strict sequencing restriction in order to provide more freedom for scheduling. Our notation for an airport with a given airport capacity curve C_{ac} during a period of time T , with the basic time unit set at 15 minutes are defined as follow:

- N_t : total number of time slots within the period T ;
- $\{t\}$: $t = 1, \dots, N_t$ are the index of the time slots;
- $r_{i,a}$: the time slot in which the arrival flight i is assigned to land on the airport;
- $r_{i,p}$: the time slot in which the arrival flight i plans to land;
- $d_{j,a}$: the time slot in which the departure flight j is assigned to take off;
- $d_{j,p}$: the time slot in which the departure flight j plans to take off;
- N_r : The total number of arrival flights within T ;
- N_d : The total number of departure flights within T ;
- c_{air} : Airborne holding cost per 15 min;
- c_{gnd} : Ground holding cost per 15 min;
- $R(t)$: The number of arrival flights being assigned to land at the time slot t ;
- $D(t)$: The number of departure flights being assigned to take off at the time slot t ;
- $C_{ac}(R(t))$: A function returning the value of the airport departure capacity according to the given arrival capacity.

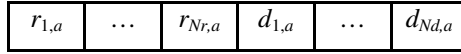


Fig. 2. The Chromosome Structure Used in the Problem

The total cost for delay within the time period T is then:

$$f = \sum_{i=1}^{N_r} c_{air}(r_{i,a} - r_{i,p}) + \sum_{j=1}^{N_d} c_{gnd}(d_{j,a} - d_{j,p}) \tag{5}$$

Given $\{r_{i,p}\}, i = 1, \dots, N_r$ and $\{d_{j,p}\}, j = 1, \dots, N_d$, our objective is to find the values of $\{r_{i,a}\}, i = 1, \dots, N_r$ and $\{d_{j,a}\}, j = 1, \dots, N_d$ in order to minimize the cost f , subject to the constraint $D(t) \leq C_{ac}(R(t))$ for all $t \in T$. Note that it is unreasonable to allow a flight to take off or land in a time slot before the original planned schedule. Therefore, the assignment of a flight to a time slot earlier than it is planned is forbidden. Since $(r_{i,a} - r_{i,p})$ and $(d_{j,a} - d_{j,p})$ are always non-negative, f is also always non-negative.

We apply SAMOGA to optimize a simulated schedule at the Beijing International Airport within a period T of 10 hours. First, we divide the 10-hour-period into 15-minute intervals, which gives a total of $N_t = 40$ time slots. Then we generate flights schedules $\{r_{i,p}\}$ and $\{d_{j,p}\}$, with $N_r = N_d = 250$, and $c_{air} = 2$ and $c_{gnd} = 1$. The airport capacity curve is given by Fig.1. The chromosome structure used is shown in Fig.2. The chromosome simply combines arrival flights with departure flights to give a chromosome with length $L = N_r + N_d$. In order to increase the efficiency, we use a repair scheme to improve the quality of the chromosomes before they are mutated. In order to make full use of the available capacity of the airport, when the point $(R(t), D(t))$ is inside C_{ac} , we will reduce the number of delayed slots of the delayed flights by assigning them to the time slot t one by one, till the point $(R(t), D(t))$ is on the capacity curve C_{ac} or no more delayed flights are available for the assignment without violating other constraints. In this repair scheme, the arrival flights are always reassigned first due to its higher priority over departure flights. On the other hand, if a chromosome is found to have violated any constraints (e.g. exceeding the capacity), and if the problem is still not fixed after the repair scheme, the chromosome will be given a penalty on its fitness and it will have a much lower rank in the population matrix. Through mutation, this unfit chromosome will quickly be eliminated. Our population size is 150. MOGAR, MOGAC and MOGAM (with preset time sharing parameter $\gamma = 0.5$) are also performed on the same problem, using the same set of initial conditions and initial population.

5 Results

We compare the performance of four kinds of genetic algorithms: MOGAR, MOGAC, MOGAM($\gamma=0.5$), and SAMOGA in Fig.3. The fitness (flight delay cost) of the best chromosome for the first 150 generations is shown here. From our graph, we can see that MOGAC has shown its weakness of early convergence and it fails to further reduce the delay cost after 50 generations. On the other hand, MOGAR, which emphasizes the exploration of the solution space, has a slow convergence: it is the slowest to optimize the delay cost to a value lower than 200. It also fails to reach the

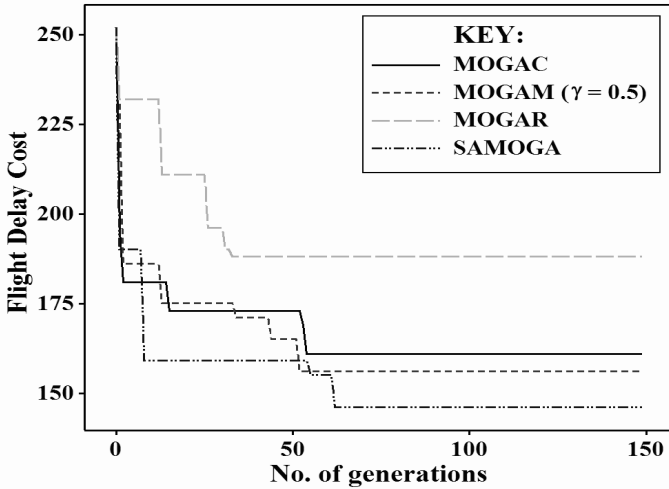


Fig. 3. Graph Showing the Performances of 4 Mutation Only Genetic Algorithms on the Same Problem for the First 150 Generations

optimum within the first 150 generations. Now, MOGAM with the time sharing parameter $\gamma = 0.5$ has equal time sharing of MOGAR and MOGAC. It shows better performance compared to MOGAR and MOGAC alone. Finally, SAMOGA, being a parameter-free combined algorithm of MOGAR and MOGAC, demonstrates an efficient and reliable approach to the optimal solution. It outperforms the MOGAR and MOGAC shortly after 40 generations. From Fig.3 we can see that the approach towards the optimal solution is quite steady without the problem of early or slow convergence. It also outperforms MOGAM after 60 generations and achieves the lowest flight delay cost. We have verified these conclusions in other simulations and we can safely conclude that SAMOGA is a better genetic algorithm compared to the other three, and with the additional the advantage of being parameter-free.

6 Conclusion

In this paper we develop a parameter free evolutionary algorithm based on idea of portfolio management on the temporal resource allocation of the CPU time to two competing algorithms: MOGAR that emphasize exploration and MOGAC that emphasizes exploitation. As MOGAR and MOGAC are themselves using adaptive parameter control via the mutation matrix, we see that our general philosophy of parameter free genetic algorithm is indeed very effective in the evolutionary process for optimization of known function. Here we demonstrate the advantage of adaptive parameter control through an application to the airport scheduling problem, but other optimization problems with constraints can also be solved more effectively with this adaptive method. Our SAMOGA makes use of the performance–volatility ratio in economics and its success demonstrates the advantage of this interdisciplinary approach. Although our present method uses mutation as the only genetic operator,

extension to include crossover operator has been done for MOGA. The results are consistent with our present approach, showing the power of adaptive parameter control [7].

Acknowledgment. K.Y. Szeto acknowledges the support of grants CERG 602507 and DAG05.06/SC32.

References

1. Holland, J.H.: Adaptive in natural and artificial system. University of Michigan, Ann Arbor (1975)
2. Goldberg, D.E.: Genetic algorithms in Search, Optimization, and Machine Learning. Addison-Wesley, Reading (1989)
3. Ma, C.W., Szeto, K.Y.: Locus Oriented Adaptive Genetic Algorithm: Application to the Zero/One Knapsack Problem. In: Proceeding of The 5th International Conference on Recent Advances in Soft Computing, RASC 2004, Nottingham, U.K, pp. 410–415 (2004)
4. Li, S.P., Szeto, K.Y.: Cryptoarithmetic problem using parallel Genetic Algorithms. In: 5th International Conference on Soft Computing, Mendl 1999, Brno University of Technology, Czech, June 9-12, 1999, pp. 82–87 (1999)
5. Zhao, S.Y., Szeto, K.Y.: preprint (2008)
6. Szeto, K.Y., Zhang, J.: Adaptive Genetic Algorithm and Quasi-parallel Genetic Algorithm: Application to Knapsack Problem. In: Lirkov, I., Margenov, S., Waśniewski, J. (eds.) LSSC 2005. LNCS, vol. 3743, pp. 189–196. Springer, Heidelberg (2006)
7. Law, N.L., Szeto, K.Y.: Adaptive Genetic Algorithm with Mutation and Crossover Matrices. In: Proceedings of the 12th International Joint Conference on Artificial Intelligence (IJCAI 2007) (Volume II) Theme: AI and Its Benefits to Society, International Joint Conferences on Artificial Intelligence, IJCAI-0, Hyderabad, India, January 6-12, 2007, pp. 2330–2333 (2007)
8. Markowitz, H.: Portfolio Selection. In: Journal of Finance, vol. 7(1), pp. 77–91. Blackwell Publishing, Oxford (1952)
9. Gilbo, E.P.: Optimizing Airport Capacity Utilization in Air Traffic Flow Management Subject to Constraints at Arrival and Departure Fixes. IEEE Transactions on Control Systems Technology, 490–503 (1997)
10. Vermorel, J., Mohri, M.: Multi-Armed Bandit Algorithms and Empirical Valuation. In: Gama, J., Camacho, R., Brazdil, P.B., Jorge, A.M., Torgo, L. (eds.) ECML 2005. LNCS (LNAI), vol. 3720, pp. 437–448. Springer, Heidelberg (2005)
11. Jinling, J., Ding, J., Wang, H.: Optimization of Airport Flight Arrival and Departure Based on Compromise Immune Algorithm. In: ICNC (2007)

Cross Checking Rules to Improve Consistency between UML Static Diagram and Dynamic Diagram

Ilkyu Ha and Byunguk Kang

Department of Computer Engineering, Yeungnam University
Kyungsan, Kyungbuk 712-749, Republic of Korea
ilkyuha@yumail.ac.kr
bwkang@yu.ac.kr

Abstract. There are many well-formedness rules of each UML element in UML specification[1], but there are not any rules that check the consistency among UML diagrams. Therefore, in this paper, we propose several checking rules to improve the consistency among UML diagrams, especially between UML static diagram and dynamic diagram. So we make explicit some requirements on consistency of UML diagrams that are buried in the original well-formedness rules of UML specification and derive some checking rules. Finally, we examine the usefulness of the derived rules through a case study.

1 Introduction

The UML(Unified Modeling Language) is a widely accepted standard in object-oriented modeling. Because the UML is semantically rich, the target system can be described in detail, but it is hard to assure the rightness of designed diagrams. The impact of erroneous diagrams might be expanded in software process. Therefore, it is important to minimize errors by verifying user diagrams in early design stages. There are several characters for the rightness of UML diagrams: completeness, consistency and correctness. The completeness is a character for checking whether user requirements are completely reflected on nine diagrams. The consistency is a character for checking whether nine diagrams are coherently designed according to the requirements. The correctness is a character for checking whether user diagrams are suitable for the UML standards.

In this paper, focusing on the consistency among three characters, we propose several verification rules for checking consistency between UML static diagram and dynamic diagram. As the process, first, the metamodels of static diagram and dynamic diagram are built from the metamodel of nine UML diagrams. Second, consistency rules for checking consistency are derived from the metamodels. The derived rules are specified formally with a specification language such as OCL(Object Constraint Language)[2]. Finally, the usefulness of the derived rules is evaluated through a case study.

2 Derivation of Metamodels and Analysis of Relationships

2.1 Related Works

There are several methods to verify consistency between UML diagrams: meta-model-based method[3], graph-based method[4-6], scenario-based method[7] and

constraints-based method[8-10]. The metamodel-based method derives the meta-models of UML diagrams and checks the consistency using the relationship of metamodels. The graph-based method uses some graphs as a conversion form of UML diagram and checks the consistency between graphs by applying some special grammars. The scenario-based method compares static information to the dynamic information of UML diagrams, and checks whether the static information is satisfied with the dynamic information. The constraints-based method uses the constraints that are applied to model. This method expresses an object model with its own special specification language and applies constraints to the model for consistency checking.

These methods have some problems. First, the verification rules for checking syntactical correctness are not classified methodically and systematically. Especially the consistency rules for checking consistency between diagrams are not sufficient. Second, these methods use some specification languages that are different from OCL. Therefore, in some cases, they are not proper to specify UML elements and the relationships between UML elements.

2.2 Features of the Proposed Rules

The proposed rules have several distinctive features in comparison with UML specification. First, the proposed rules focus on the consistency among 9 UML diagrams, especially between static diagram and dynamic diagram. We derive the checking rules from some requirements on consistency of UML diagrams that are buried in the original UML specification. Second, the proposed rules are specified with OCL, the standard constraints language of UML. Table 1 shows comparison between UML specification and the proposed rules.

Table 1. Comparison of the proposed rules to the UML well-formedness-rules

Comparison	The UML well-formedness-rules	The proposed rules
metamodel type	Package (Core, Data Types, etc)	Diagram (Static, Dynamic)
checking object	UML elements	Consistency between diagrams
number of rules	202 Well-Formedness Rules (for all UML elements)	7 Consistency Rules (for object diagram only)
Specification method	OCL	OCL

The derivation process of the proposed rules and the checking process of user diagrams are shown in Fig. 1. Firstly, the metamodels of nine UML diagrams are derived and the upper metamodels of static diagram and dynamic diagram are derived with reference to the nine UML metamodels. Secondly, some consistency rules for checking consistency are derived from relationships in metamodels. The derived rules are specified formally with OCL specification language for clear understanding and inserted into a verification tool. Finally, the consistency between static diagram and dynamic diagram is checked by applying the derived rules to user diagrams.

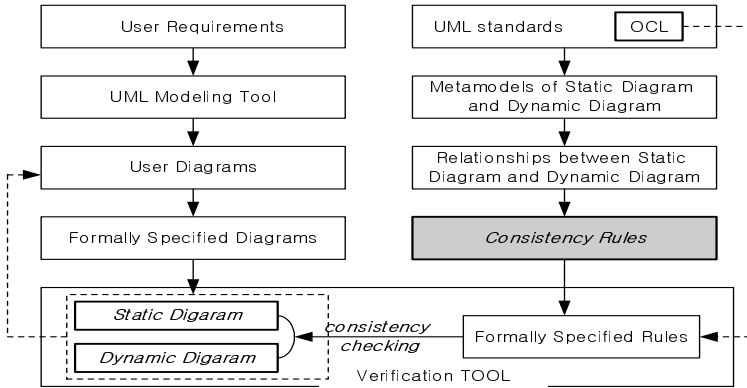


Fig. 1. The derivation process of the consistency rules

2.3 Derivation of Metamodels

The metamodels of static diagram and dynamic diagram can be derived from the metamodel of each UML diagram: the metamodel of static diagram and dynamic diagram is a metamodel of metamodel. Fig. 2 shows the metamodels of static diagram and dynamic diagram. The metamodels are used to extract the relationships among UML diagrams.

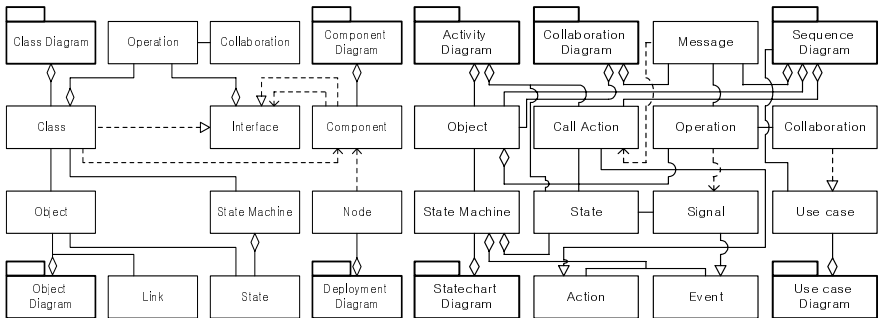


Fig. 2. Metamodels of static diagram and dynamic diagram

2.4 Analysis of the Relationships from Metamodels

The metamodels include common elements such as classes, objects and states among diagrams. And they include relationships such as generalization, association and dependency among the elements. We can see whether a relationship exists between diagrams from the metamodels. So we can make a relation graph from the metamodels as Fig. 3. The graph presents relationships among nine UML diagrams.

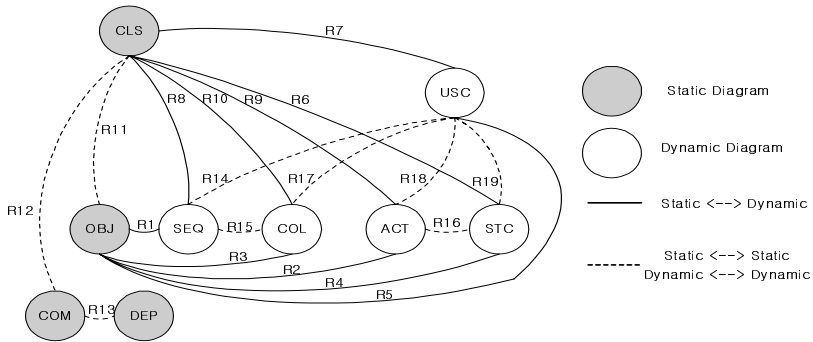


Fig. 3. Relationships between nine UML diagrams

The relation R1 shows a relationship between object diagram and sequence diagram. The object diagram involves modeling a snapshot of the system at a moment in time and rendering a set of objects, their states and their relationships. As the object diagram is an instance of class diagram, object, operation and link of object diagram have association relationships with object, message and link of sequence diagram as relation R8. The relation R2 shows a relationship between object diagram and activity diagram. There is an association relationship between object of object diagram and object of activity diagram as relation R9. The relation R3 shows a relationship between object diagram and collaboration diagram. The relationships between object diagram and dynamic diagrams are derived as Table 2.

Table 2. Relationships between object diagram and dynamic diagrams

relation number	Static Diagram		relationship	Dynamic Diagram	
	diagram	element		element	diagram
R1	object	object	correspondence	object	sequence
	object	operation	correspondence	message(call)	sequence
	object	link	correspondence	link	sequence
R2	object	object	correspondence	object	activity
R3	object	object	correspondence	object	collaboration
	object	operation	correspondence	message	collaboration
	object	link	correspondence	link	collaboration

3 Derivation of Consistency Rules

The consistency rules between object diagram and dynamic diagrams are derived from the relationships of table 2. These rules check generally whether a diagram has a correct element that corresponds to an element in corresponding diagram. Fig. 4 shows that an object diagram must have an object that corresponds to an object in a sequence diagram. Therefore we can derive the consistency rules as R1-1.

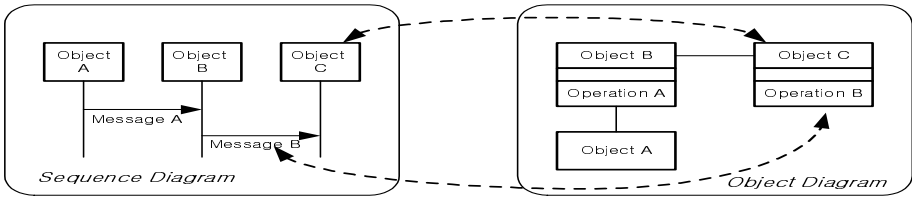


Fig. 4. Relationships between object diagram and sequence diagram

R1-1. The object diagram must have an object that corresponds to an object in the sequence diagram that has relation to itself.

There should exist an operation of the called object related to the message that generated from call action in sequence diagram as the relation R1 in table 2. Therefore, a consistency rule is derived as R1-2.

R1-2. The object diagram must have an operation of an object that corresponds to a message in the sequence diagram that has relation to itself

If there is a link between objects in sequence diagram, there should exist a link between objects in the object diagram that has relation to the sequence diagram. Therefore, a consistency rule is derived as R1-3.

R1-3. An object diagram must have a link that corresponds to a message link in the sequence diagram that has relation to itself.

An activity diagram is used to model the behavior to which object can take within several use cases. At this time an object in activity diagram has relation to an object in an object diagram, and a rule is derived as R2-1.

R2-1. The object diagram must have an object that corresponds to an object in activity diagram that has relation to itself.

A collaboration diagram is an interaction diagram that emphasizes the static organization of the objects that send and receive messages. The collaboration diagram has object, link and message as the sequence diagram does. Therefore the consistency rules between object diagram and collaboration diagram are derived as R3-1~R3-3.

R3-1. The object diagram must have an object that corresponds to an object in the collaboration diagram that has relation to itself.

R3-2. The object diagram must have an operation of an object that corresponds to a message in the collaboration diagram that has relation to itself.

R3-3. The object diagram must have a link that corresponds to a message in the collaboration diagram that has relation to itself.

4 Usefulness of the Consistency Rules

4.1 Formal Specification of the Derived Rules

If the derived rules are expressed with a formal language, the consistency of the diagrams could be checked easily and automatically. Therefore the rules need to be

changed in some formal types. So in this paper, the OCL is used to specify the rules. The OCL is an expression language that enables one to describe constraints on object-oriented models. There are three standard stereotypes in UML constraints: invariant, pre-condition and post-condition[11]. In this paper, the rules are described with invariant type as follows.

```
context verified_element inv rule_name:
    verification_condition
```

The *verified_element* is an element in the diagram that will be verified, the *rule_name* is a name that is given to each rule, and the *verification_condition* is an expression of the verification rule. The *verification_condition* decides whether elements are suitably expressed for requirements, and it usually returns true or false value. For example the R1-1 is specified with the previous type as follows.

```
Contents1():Set(Object_b)=self.r_objectb;
Matchparentb():Set(DynamicDiagramElement)=self.r_dynamicdiagramelement;
Matchparentd():Set(DiagramElement)=self.Matchparentb()->collect(clc.parent()->flatten->asSet;
Matchchilds():Set(StaticDiagramElement)=self.Matchparentd()->collect(clc.childs()->flatten->asSet;
Matchchildobj():Set(ObjectDiagramElement)=self.Matchchilds()->collect(clc.childobj()->flatten->asSet;
Matchchildobj():Set(Object)=self.Matchchildobj()->collect(clc.childobj()->flatten->asSet;
Matchseqd(ob:Object_b):Boolean=(self.Matchchildobj()->exists(clc.object_name=ob.object_name));
context SequenceDiagramElement inv R1-1:
self.Contents1()->forAll(obself.Matchseqd(ob))
```

The *Contents1()* extracts all of the objects from sequence diagram and the *Matchseqd()* decides whether there is an object in object diagram that has relation to the object.

4.2 Checking Consistency between Example Diagrams

The OCL-formed rules can be interpreted using OCL compilation tool[12-13]. As the OCL is a specification language that has formal grammar, it is easy to automate the interpretation of OCL specification. There are several tools related to OCL compilation now. One of the most distinguished tool is the USE[14]. Therefore, we use the USE tool for verifying the usefulness of our consistency rules.

For examining the usefulness of the derived consistency rules, two examples are introduced as Fig. 5. The left side is an object diagram that shows relationships among

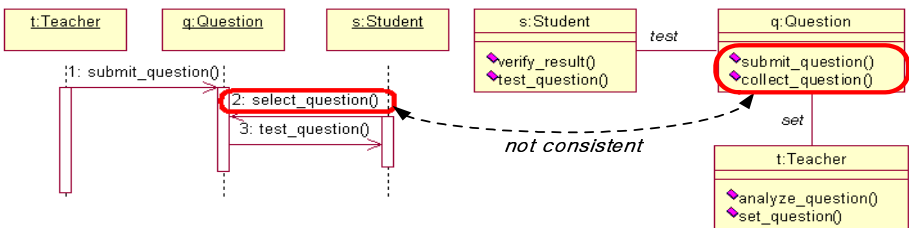


Fig. 5. Examples of object diagram and sequence diagram

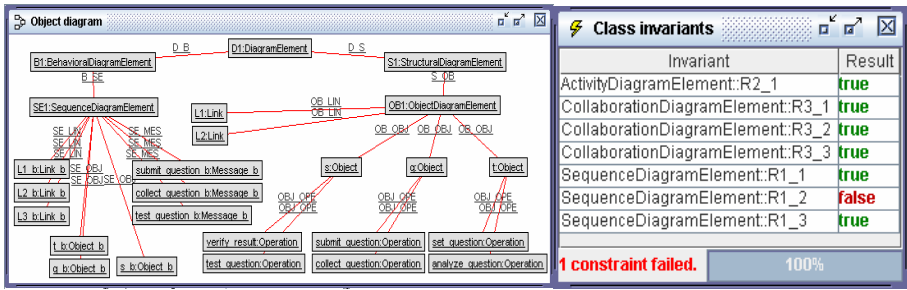


Fig. 6. A created instance model and verification results

‘student’, ‘question’ and ‘teacher’ objects in a learning and evaluation system. The right side is a sequence diagram that corresponds to the object diagram. The sequence diagram has an error, namely it includes an incorrect message as the signed part. As the rule R1-2, an object diagram must have an operation of an object corresponded to a message in sequence diagram, but the corresponded operation is not in the object in object diagram.

The above diagrams are converted into an instance model by execution of USE commands. The left side of Fig. 6 shows the instance model of the two diagrams. The right side of Fig. 6 shows the result of the validation. The diagrams of Fig. 5 have a fault in their relationships. Namely, they are not satisfied with the R1-2 consistency rule, therefore the result of the rule R1-2 shows a ‘false’.

5 Conclusions

In this paper, we proposed several checking rules to improve the consistency among nine UML diagrams. The derived rules were especially focused on the consistency between UML static diagram and dynamic diagram. While the well-formedness rules of UML specification were focused on correctness of UML elements, we made explicit some requirements on consistency of UML diagrams that are buried in the original well-formedness rules of UML specification and derived some useful consistency rules.

The usefulness of the derived rules had been evaluated through a case study. After evaluation, we were convinced that the rules are so useful for checking consistency between UML diagrams. When we inserted some faults into example diagrams and checked the diagrams with the consistency rules, the rules detected faults about consistency successfully. And we are convinced that the rules can be applied to UML modeling tools as a new checking rule to improve the consistency between diagrams.

References

1. OMG: OMG Unified Modeling Language Specification Version. 2.0 Object Management Group Inc. (2005)
2. OMG: Object Constraint Language Specification in OMG Unified Modeling Language Specification Version 2.0 Object Management Group Inc. (2006)

3. Chong, K.W., Cho, Y.S., Kwon, S.G.: Detecting Errors and Checking Consistency in the Object-Oriented Design Models. *Journal of KIPS*, 2072–2087 (1999)
4. Tsiolakis, A., Ehrig, H.: Consistency analysis of UML class and sequence Diagrams using attributed graph grammars. In: *Proc. of Joint APPLIGRAPH/GETGRATS Workshop* (2000)
5. Tsiolakis, A.: Consistency Analysis of UML Class and Sequence Diagrams based on Attributed Typed Graphs and their Transformation. Technical Report 2000/3. Technical University of Berlin (2000)
6. Sunetnanta, T., Finkelsteing, A.: Automated Consistency Checking for Multiperspective Software Specifications. In: *Proc. of ICSE 2001* (2001)
7. Cho, J.H., Bae, D.H.: Scenario-Driven Verification Method for Completeness and Consistency Checking of UML Object-Oriented Analysis Model. *Journal of KISS* (2001)
8. Richters, M.: A Precise Approach to Validating UML Models and OCL Constraints. PhD thesis. University Bremen. Logos Verlag, Berlin. BISS Monographs. No.14 (2002)
9. Richters, M., Gogolla, M.: Validating UML models and OCL Constraints. In: Evans, A., Kent, S., Selic, B. (eds.) *UML 2000. LNCS*, vol. 1939, pp. 265–277. Springer, Heidelberg (2000)
10. Bottoni, P., Koch, M., Parisi-Presicce, F., Taentzer, G.: Consistency Checking and Visualization of OCL Constraints. In: Evans, A., Kent, S., Selic, B. (eds.) *UML 2000. LNCS*, vol. 1939. Springer, Heidelberg (2000)
11. Wormer, J.B., Kleppe, A.G.: *The Object Constraint Language*. Addison-Wesley, Reading (1999)
12. Hussmann, H., Demuth, B., Finger, F.: Modular architecture for a toolset supporting OCL. In: Evans, A., Kent, S., Selic, B. (eds.) *UML 2000. LNCS*, vol. 1939. Springer, Heidelberg (2000)
13. IBM: OCL Parser ver.o.3. (2005), <http://www-3.ibm.com/software/ad/library/standards/ocl.html>
14. Richters, M.: The USE tool: A UML-based specification environment (2001), <http://www.db.informatik.uni-bremen.de/projects/USE/>

Neural Networks Approach to the Detection of Weekly Seasonality in Stock Trading

Virgilijus Sakalauskas and Dalia Kriksciuniene

Department of Informatics, Vilnius University, Muitines 8, 44280 Kaunas, Lithuania
{virgilijus.sakalauskas,dalia.kriksciuniene}@vukhf.lt

Abstract. In this article we investigate the problem of detection the statistically significant dependences of stock trading return, which occur in particular days of the week (usually the first or the last trading day), and which could be important for creating profitable investment strategies. The identifying such days of the week (day-of-the-week effect) is performed by using artificial neural networks. The research results helped to conclude the effectiveness of application of neural networks, as compared to the traditional linear statistical methods for finding stock trading anomalies. The effectiveness of the method was confirmed by exploring impact of different variables to the day-of-the-week effect, evaluation of their influence and sensitivity analysis, and by selecting best performing neural network type. The experimental verification was implemented by using Vilnius Stock Exchange trading data.

Keywords: artificial neural networks, day-of-the-week anomaly, MLP, mean return, RBF, stock market.

1 Introduction

The efficient market hypothesis states that it is not possible to consistently outperform the market by using any information that the market already knows. However, researchers have reported evidence of abnormal returns related to day of the week effects, also week of the month, month of the year, turn of the month, turn of the year effects or holiday effects. In the extensive study of market anomalies, based on analysis of Dow Jones daily data, registered during 100 years period, and of about 25 years of S&P500 daily data, the set of rules, which contained nearly 9,500 different calendar effects was tested [1]. Mostly significantly different stock returns come through the first and the last trading days of a week. As indicated in [2-4] daily stock returns tend to be lower on Mondays and higher on Fridays for the United States and Canada. Therefore, methods of predicting any divergences from the efficient market hypothesis may allow to develop profitable trading strategies and to decrease the investment risk.

The day-of-the-week (DOW) effect is understood as significantly different stock returns in different days of a week. It means that different trade situation in different days of a week have some features of regularity. A number of scientific papers, devoted to analysis of this anomaly [5-8], have disclosed that from 1980 DOW effect was clearly evident in the vast majority of developed markets, but it appears to have

faded away in the 1990s [9,10]. This implication is based on long-run improvements in market efficiency, which may have lowered the effects of certain anomalies in recent periods [4]. The other reason of the diminished significance of the calendar effects is the inability of the prevalent statistical methods to recognize them. As it is noticed in [1], the anomalies are most outstanding in short periods and in comparatively small data subsets. The market inefficiency, as investigated in [11], could be observed by applying research of higher moments instead of traditional statistical analysis, and also taking into account all available information about the stocks: the trading volumes, dividends, earnings-price ratios, prices of other assets, interest rates, exchange rates, and, subsequently, usage of powerful nonparametric regression techniques, such as artificial neural networks.

The main variable, used for DOW effect investigation in the research literature is the mean return. The other important indicators for the research are the daily closing prices and indexes [12]. However, application of other variables for establishing this anomaly is quite rare. In the paper we apply the artificial neural networks (ANN). The researched data includes mean return and also number of deals and shares, turnover, H-L (high minus low price).

There are few research works of applying ANN for DOW effect analysis or to the related questions of the efficient market hypothesis. Yet the results obtained are quite encouraging to use them for further research.

Neural networks are an artificial intelligence method for modelling complex target functions. Recently artificial neural networks methods have started to be used widely to the domain of financial time series prediction. However, it is a highly complicated task. Financial time series often behave nearly like a random-walk process and the predictability of stock prices or levels of indices under the efficient market hypothesis is impossible. On the other hand, statistical behaviour of the financial time series is different at different points in time and usually is very noisy.

ANN mostly are used for predicting stock performance [13,14], classification of stocks, predicting price changes of stock indexes [15], forecasting the performance of stock prices [16-18]. In most analyzed applications, the neural network (NN) results outperform statistical methods, such as multiple linear regression analysis, ANOVA, discriminant analysis and others [19].

For investigation the DOW effect we apply artificial neural networks classification potential. Two standard types of neural networks were used: MLP (Multilayer Perceptrons) and RBF (Radial Basis Function Networks). The research outcomes and conclusions were evaluated, and revealed better effectiveness of the neural networks than traditional statistical analysis in identifying DOW anomaly.

2 Data and Methodology

In the paper data for empirical research was taken from information base of Vilnius Stock Exchange (The Nordic Exchange, 2008 [20]). Vilnius Stock Exchange belongs to the category of small emerging securities markets. It is proved by the main financial indicators: market value of 7 billions EUR, near 2 million EUR share trading value per business day, approximately 600 numbers of trades per business day, and the equity list consisting of 44 shares.

To identify the presence of DOW effect we shall use the data of daily return values of 24 shares (out of 44 listed), from period of 2003-01-01 till 2006-11-21. The selected shares represent variety of Vilnius Stock Exchange equity list by capitalization, daily turnover, trade volume, return and risk, and thus ensure the validity of the research results.

In the paper will be used logarithmic understanding of return (1), where return on time moment t , R_t is evaluated by logarithmic difference of stock price over time interval $(t-1,t]$.

$$R_t = \ln\left(\frac{P_t}{P_{t-1}}\right), \quad (1)$$

where P_t indicates the price of financial instrument at time moment t .

The return values of 24 equities were assigned to the variables, named correspondingly to their symbolic notation of Vilnius Stock Exchange. The initial analysis by summary statistics of the data set, revealed quite big differences in return for different days of a week.

Application of traditional statistical methods, such as t-test and one-way ANOVA, Levene and Brown-Forsythe test of homogeneity of variances confirmed the DOW effect only for few stocks out of the total 24 [21].

Using Kolmogorov-Smirnov test for different days of the week, and to all possible pairs of the day it was defined that significant difference for some days of the week could be confirmed only for 3 variables, namely LDJ, PZV, UTR (out of 24) [21].

The traditional statistical analysis could make some indications of the DOW effect by finding differences in mean values of return of securities, but did not allow to state that these differences are significant.

Further for research of this anomaly we will use DOW classification method with the help of artificial neural networks.

To determine the DOW effect traditional statistical methods are usually used and the only most important variable - mean return - is invoked. However, in the case of vaguely expressed DOW anomaly, such approach is not sufficient. For the analysis we should invoke more powerful nonlinear statistical research and use more variables that may have impact for the DOW effect. Here we used artificial neural networks for exploring presence of the DOW effect for the same data set of 24 equities from Vilnius Stock Exchange. The idea of research was to attempt distinguish Monday/Friday from other days of the week using neural networks classification potential.

In our research we shall apply two standard types of neural networks, implemented in the STATISTICA Neural Networks standard module: MLP (Multilayer Perceptrons) and RBF (Radial Basis Function Networks), evaluated for good performance of classification tasks [6].

In this research both types of neural networks, MLP and RBF, were used to distinct Monday and Friday from the other trading days of the week. We shall use the predicting variables: mean return, number of deals and shares, turnover, H-L (high minus low price). During the research we will attempt to select the best neural network for

classification, to choose most influential variables to the final result, and make their sensitivity analysis.

To determine the DOW effect we used from 750 to 950 items of data, accordingly to each investigated security. The non trading days of the particular securities data were eliminated from the sets.

As Monday/Friday records are 4 times less than other days of the week, classification could be made presuming all days are ‘others’ (vary from the Monday/Friday). In such a case we obtain fairly high correct classification percentage (80%). Though there would be no correctly classified Monday/Friday records at all. We will not proceed this way, as we do not have a goal to receive as high as possible general level of classification. We only wish to see if predictors can fairly significant distinguish Mondays/Fridays from other days of the week if *a priori* it is not known what day of the week is analyzed. Thus applying ANN and classifying the cases we shall presume that prior probabilities for Monday/Friday and other days of the week are equal. We also do not have a necessity to divide data to training, test and verification sets as we do not need to obtain classification instrument, appropriate for classification of future data. We only want to prove the presence of DOW anomaly using available data. For calculation we shall use the Statistica Neural Network Intelligent Problem Solver tool, which allowed to automate classification by indicating input and output variables, assign distribution cases, accept-reject confidence limits, network complexity options, set limit duration of design process, results to be displayed.

Further we explain the data processing procedure by applying ANN for one of the securities (TEO), aimed to distinct Monday from other days of the week. Same procedure was applied for each 24 securities data set.

Using Statistica Neural Network Intelligent Problem Solver tool we select DAY as output (dependent) variable and DEALS, NO_OF_SH, TURNOVER, RETURN, H_L as input (independent) variables. The performance of best found network (correct classification rate, error, area under ROC curve) is presented in generated report by Intelligent Problem Solver (Fig. 1).

The report, presented in Fig. 1, states finding best classification algorithm took 3:43 min. It resulted in finding improved MLP (3 layer) network specified as 3:3-7-1:1 (3 input variables, 7 hidden units and 1 output variable), error level achieved 0.36817. The error was calculated as the root mean square (RMS) of errors, measured for each individual case. The ST Neural Networks training algorithms attempted to minimize the training error, the final performance measure indicated rate of correctly classified cases. This result depends on Accept and Reject thresholds (confidence limits).

In Fig. 1, the correct classification rate is 0.623377, therefore 62.33% cases were correctly classified (both Monday and other days).

Time	The best network found had O.K. performance			
	Profile	Classif.rate	Error	Under ROC
0:03:43	MLP 3:3-7-1:1	0,623377	0,368174	0,676299

Fig. 1. Intelligent Problem Solver report of TEO equities Monday effect

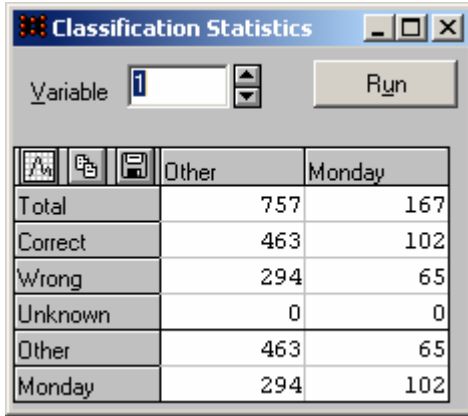


Fig. 2. Statistics of Monday data recognition

The classification statistics report (Fig. 2) presents number of correctly classified data of Monday from other trading days.

As the correct classification number highly exceeds wrongly classified data, the Monday effect for TEO security is confirmed.

In further analysis we try to define, which input variables were most important for classification, and to make the sensitivity analysis for the neural network (Fig. 3).

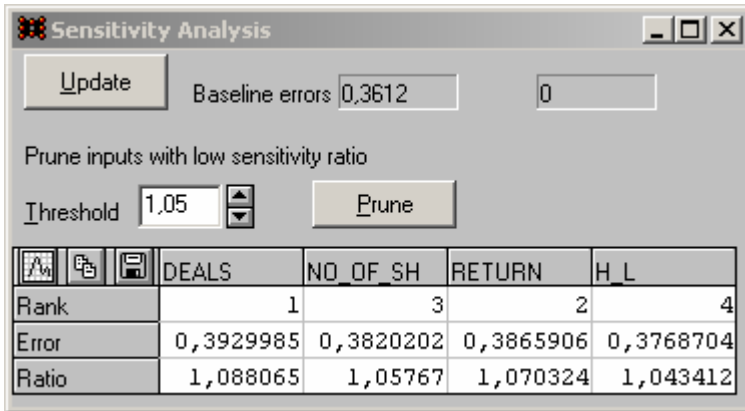


Fig. 3. Sensitivity analysis of TEO securities (Monday)

The figure shows, how classification results could be altered by rejecting each variable, e.g. if we reject variable DEALS, the baseline error becomes 0.393 (Fig. 3).

The sensitivity analysis ranked input variables, according to their impact for network performance. The highest error is assigned for the highest ranked, it shows, how network performance deteriorates, if they are not present (Fig. 3). Obviously that for different stocks other variables can have the biggest impact.

In the Statistica Neural Networks software the network performance can be evaluated according to the achieved correct classification rate as poor (rate is less than 0.6), O.K and High (over 0.8). The outcome of analysis for security (TEO) states, that it is influenced by Monday effect, and the constructed neural network distinct Mondays from the other trading days with sufficient reliability. Further we also assume classification performance as significant, if it is higher, than 0.6.

The experiment indicate that the most important variables in most cases are DEALS, H_L and RETURN (Table 1). Variables NO_OF_SH and TURNOVER mostly are not so important in researching the DOW effect.

The above described neural classification data processing procedure for all data of 24 securities, is summarized in Table 1. The significant performance values (> 0.6), which indicate presence of Monday or Friday effect for the particular securities, are highlighted with dark background.

From Table 1, we can conclude that only results for of 11 equities confirm Monday effect, and for 9 equities the Friday effect was present. Both effects were present for 4 equities. In the Table 1, the equities were sorted according to average daily trading turnover, which is traditional parameter in the research of DOW effect. However, as it may be seen from the Table 1, turnover impact for Monday/Friday anomaly may not be confirmed. As mentioned before, most important parameters having impact for the DOW particularity are the number of deals, difference between highest and lowest equity price and mean return.

Table 1. Application results (Monday and Friday effect is marked by dark background)

	Monday				Friday			
	Performance	Error	Major Var	Network Input Hidd.	Performance	Error	Major Var	Network Input Hidd.
LEL	0.5623	0.3909	H-L	MLP 2 6	0.5770	0.3849	H-L	MLP 3 6
KBL	0.6501	0.3590	Deals	MLP 5 10	0.5214	0.3994	H-L	RBF 4 9
LNS	0.6066	0.3656	No_Sh	MLP 5 7	0.5436	0.3961	H-L	MLP 3 3
LEN	0.5576	0.3815	No_Sh	MLP 3 6	0.7291	0.3590	Deals	RBF 5 180
VBL	0.5943	0.3640	Turn.	MLP 5 6	0.5814	0.3870	H-L	MLP 4 6
LJL	0.5193	0.3918	Turn.	RBF 3 10	0.6409	0.3910	Deals	RBF 5 66
LLK	0.6281	0.3683	Deals	MLP 4 6	0.6638	0.3563	H-L	MLP 5 10
KJK	0.6738	0.3526	H-L	MLP 4 7	0.5778	0.3829	H-L	RBF 4 13
ZMP	0.5776	0.3807	Return	MLP 4 7	0.6007	0.3566	H-L	MLP 4 7
LDJ	0.4850	0.3857	Deals	MLP 4 6	0.5795	0.3869	Return	RBF 3 12
RST	0.5713	0.3927	Deals	RBF 3 7	0.5245	0.3953	H-L	RBF 4 12
UTR	0.5854	0.3736	Deals	MLP 3 10	0.6161	0.3717	H-L	MLP 5 8
NDL	0.6656	0.3419	H-L	MLP 5 8	0.6748	0.3426	Deals	MLP 5 8
SAN	0.5671	0.3682	Deals	MLP 5 8	0.5855	0.3924	H-L	RBF 3 19
KNF	0.7058	0.3554	H-L	RBF 3 119	0.6671	0.3734	Deals	RBF 3 79
PTR	0.5707	0.3760	Deals	MLP 4 4	0.6820	0.3475	Deals	MLP 5 10
PZV	0.6282	0.3639	Deals	MLP 5 11	0.5734	0.3953	Return	RBF 3 18
APG	0.6453	0.3633	Deals	MLP 5 11	0.5668	0.3964	Deals	RBF 2 13
MNF	0.6346	0.3622	H-L	MLP 4 8	0.5569	0.3916	H-L	RBF3 13
SNG	0.5571	0.3936	Deals	RBF 2 7	0.5447	0.3951	H-L	RBF 2 9
UKB	0.5795	0.3818	Deals	RBF 3 17	0.5906	0.3767	Turn.	MLP 4 11
RSU	0.5709	0.3828	Deals	MLP 3 5	0.5456	0.3918	Return	MLP 4 5
LFO	0.6583	0.3720	Deals	RBF 4 34	0.6761	0.3480	Deals	MLP 6 10
TEO	0.6234	0.3682	Deals	MLP 5 11	0.5563	0.3912	H-L	MLP 3 4

From the summary table the performance of neural networks (MLP versus RBF) showed that for indicating Monday effect best results were achieved by using MLP (18 times), and the RBF performed better for 6 times. The Friday effect was indicated with the same precision by both types of neural networks (12 times each - MLP and RBF). The neural networks MLP and RDF used similar number of input variables, but the number of elements in hidden layer for RBF was much bigger.

Comparing the performance of methods of neural networks and general linear statistical analysis for investigation of the DOW effect [21] revealed the advantages of the neural computations. By applying them the presence of the DOW effect was confirmed in 20 cases, whereas application of Kolmogorov-Smirnov test could show significant indications only for 3 such cases [21].

3 Summary and Conclusions

Despite of quite extensive analysis of the day of the week effect in scientific literature, there is evidently lack of investigations of applying artificial neural networks in this sphere.

In this article day of the week effect was researched for the case of Lithuanian stock exchange, which can be assigned to the category of small emerging stock markets. Daily trade values of 24 shares of Vilnius Stock Exchange of the time interval from 2003-01-01 to 2006-11-21 were arranged for analysis using artificial neural networks.

The calculations have been made by applying functional modules and standard procedures implemented in software package Statistica Neural Network.

The investigation was based on analysis of numerous variables: return, deals, number of shares, turnover, H-L (high minus low price). Sensitivity analysis of variables showed, that the most important impact to define DOW effect was made by variables DEALS, H_L and RETURN.

By applying Statistica Neural Network software the best classifying neural networks were selected, though no preference could be given for MLP and RBF neural networks due to their similar performance. Neural networks were more effective for revealing Monday and Friday effect, comparing to the methods of traditional statistical analysis.

Application of neural networks and the sensitivity analysis revealed that there are more variables, which have significant influence for the DOW effect. The results of the research, presented in the article, reveal that the analysis should be based not only on values of daily mean return, but should use data of the variables H-L and DEALS as well.

The research results helped to conclude the effectiveness of application of neural networks, as compared to the traditional linear statistical methods for such type of classification problem, where the effect is vaguely expressed and its presence is difficult to confirm. The effectiveness of the method has been confirmed by exploring variables, influencing the DOW effect, their influence and sensitivity analysis, and by selecting best performing type of neural network.

References

1. Sullivan, R., Timmermann, A., White, H.: Dangers of Data-Driven Inference: The Case of Calendar Effects in Stock Returns, Working Paper, University of California, San Diego (1998), <http://ucsd.edu/hwcv081a.pdf>
2. Balaban, E., Bayar, A., Kan, O.B.: Stock returns, seasonality and asymmetric conditional volatility in World Equity Markets. *Applied Economics Letters* 8, 263–268 (2001)
3. Flannery, M.J., Protopapadakis, A.A.: From T-bills to common stocks: investigating the generality of intra-week return seasonality. *Journal of Finance* 43, 431–450 (1988)
4. Kohers, G., Kohers, N., Pandey, V., Kohers, T.: The disappearing day-of-the-week effect in the world's largest equity markets. *Applied Economics Letters* 11, 167–171 (2004)
5. Brooks, C., Persaud, G.: Seasonality in Southeast Asian stock markets: some new evidence on day-of-the-week effects. *Applied Economics Letters* 8, 155–158 (2001)
6. StatSoft Inc. *Electronic Statistics Textbook*. Tulsa, OK: StatSoft. WEB (2006), <http://www.statsoft.com/textbook/stathome.html>
7. Tang, G.Y.N.: Day-of-the-week effect on skewness and kurtosis: a direct test and portfolio effect. *The European Journal of Finance* 2, 333–351 (1998)
8. Sakalauskas, V., Kriksciuniene, D.: The Impact of Taxes on Intra-Week Stock Return Seasonality. In: Bubak, M., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2008, Part II. LNCS, vol. 5102, pp. 504–513. Springer, Heidelberg (2008)
9. Kamath, R., Chusanachoti, J.: An investigation of the day-of-the-week effect in Korea: has the anomalous effect vanished in the 1990s? *International Journal of Business* 7, 47–62 (2002)
10. Steeley, J.M.: A note on information seasonality and the disappearance of the weekend effect in the UK stock market. *Journal of Banking and Finance* 25, 1941–1956 (2001)
11. Reschenhofer, E.: Unexpected Features of Financial Time Series: Higher-Order Anomalies and Predictability. *Journal of Data Science* 2, 1–15 (2004)
12. Basher, S.A., Sadorsky, P.: Day-of-the-week effects in emerging stock markets. In: *Applied Economics Letters*, vol. 13, pp. 621–628. Taylor and Francis, Abington (2006)
13. Kumar, M., Thenmozhi, M.: Forecasting Nifty Index Futures Returns using Neural Network and ARIMA Models, *Financial Engineering and Applications* (2004)
14. Virili, F., Reisleben, B.: Nonstationarity and data preprocessing for neural network predictions of an economic time series. In: *Proc. Int. Joint Conference on Neural Networks*, Como, vol. 5, pp. 129–136 (2000)
15. Nekipelov, N.: An Experiment on Forecasting the Financial Markets, BaseGroup Labs. (2007), <http://www.basegroup.ru/tech/stockmarket.en.htm>
16. Gencay, R.: The predictability of security returns with simple technical trading. *Journal of Empirical Finance* 5, 347–359 (1998)
17. Qi, M.: Nonlinear predictability of stock returns using financial and economic variables. *Journal of Business and Economic Statistics* 17, 419–429 (1999)
18. Yao, J.T., Tan, C.L.: Guidelines for financial forecasting with neural networks. In: *Proc. International Conference on Neural Information Processing*, Shanghai, China, pp. 757–761 (2001)
19. Pissarenko, D.: Neural networks for financial time series prediction: Overview over recent research. BSc thesis (2002)
20. The Nordic Exchange, (2008), <http://www.baltic.omxnordicexchange.com/>
21. Sakalauskas, V., Kriksciuniene, D.: The impact of daily trade volume on the day-of-the-week effect in emerging stock markets. *Information Technology and Control* 36(1A), 152–158 (2007)

Bregman Divergences and the Self Organising Map

Eunsong Jang¹, Colin Fyfe², and Hanseok Ko¹

¹ Intelligent Signal Processing Lab,
Korea University, Korea

{esjang, hsko}@ispl.korea.ac.kr

² School of Computing,
University of the West of Scotland, UK
colin.fyfe@uws.ac.uk

Abstract. We discuss Bregman divergences and the very close relationship between a class of these divergences and the regular family of exponential distributions before applying them to various topology preserving dimension reducing algorithms. We apply these methods to identification of structure in magnetic resonance images of the brain and show that different divergences reveal different structure in these images.

1 Introduction

Visualizing high dimensional data is problematic since we are not equipped with senses appropriate for this task. Indeed, even the task of converting two dimensional representations on our retina to the three dimensional representations our brain makes of the world is an inverse problem which is impossible to solve with 100% accuracy. Therefore we search for low dimensional representations of high dimensional data which capture some intrinsically interesting properties of the data. One such property is capturing the local distances in the high dimensional data and trying to maintain these relationships in a low dimensional projection of the data [5].

Bregman divergences have recently received a great deal of interest recently in terms of clustering and finding unsupervised projections of a data set [2,7,6,3,1]. In this paper, we investigate the use of Bregman divergences in the creation of self-organising maps and show that different structure is revealed by using different divergences within these algorithms.

2 Bregman Divergences

Consider a strictly convex function $F : S \rightarrow \Re$ defined on a convex set $S \subset \Re^d$. A Bregman divergence between two elements, p and q , of S is defined to be

$$d_F(p, q) = F(p) - F(q) - \langle (p - q), \nabla F(q) \rangle \quad (1)$$

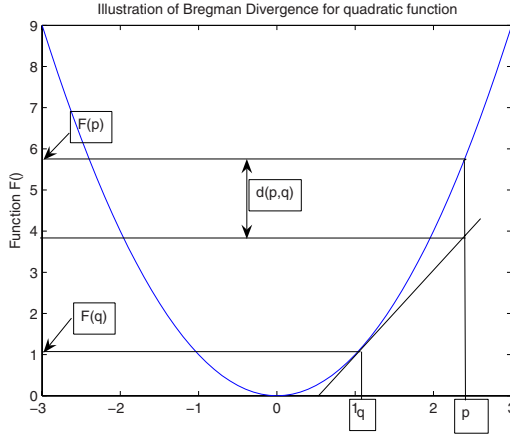


Fig. 1. The divergence is the difference between $F(p)$ and the value of $F(q) + (p - q)\nabla F(q)$

where the angled brackets indicate an inner product and $\nabla F(q)$ is the derivative of F evaluated at q . This can be viewed as the difference between $F(p)$ and its truncated Taylor series expansion around q . Thus it can be used to ‘measure’ the convexity of F : Figure 1 illustrates how the Bregman divergence is the difference between $F(p)$ and the value which would be reached from $F(q)$ with a linear change for $\nabla F(q)$.

Example 1. The squared Euclidean distance is a special case of the Bregman divergence in which $F(\cdot) = \|\cdot\|^2$

$$\begin{aligned} d_F(\mathbf{x}, \mathbf{y}) &= \|\mathbf{x}\|^2 - \|\mathbf{y}\|^2 - \langle \mathbf{x} - \mathbf{y}, \nabla F(\mathbf{y}) \rangle \\ &= \|\mathbf{x}\|^2 - \|\mathbf{y}\|^2 - \langle \mathbf{x} - \mathbf{y}, 2\mathbf{y} \rangle \\ &= \|\mathbf{x} - \mathbf{y}\|^2 \end{aligned}$$

Example 2. The Kullback-Leibler divergence is another special case in which $F(\mathbf{p}) = \sum_{j=1}^d p_j \log p_j$. Consider two discrete probability distributions, \mathbf{p} and \mathbf{q} .

$$\begin{aligned} d_F(\mathbf{p}, \mathbf{q}) &= \sum_{j=1}^d p_j \log_2 p_j - \sum_{j=1}^d q_j \log_2 q_j - \langle \mathbf{p} - \mathbf{q}, \nabla F(\mathbf{q}) \rangle \\ &= \sum_{j=1}^d p_j \log_2 p_j - \sum_{j=1}^d q_j \log_2 q_j \\ &\quad - \sum_{j=1}^d (p_j - q_j)(\log_2 q_j + \log_2 e) \end{aligned}$$

$$\begin{aligned}
 &= \sum_{j=1}^d p_j \log_2 \frac{p_j}{q_j} - \log_2 e \sum_{j=1}^d (p_j - q_j) \\
 &= \sum_{j=1}^d p_j \log_2 \frac{p_j}{q_j} = K.L.(\mathbf{p} \parallel \mathbf{q})
 \end{aligned}$$

since $\sum_{j=1}^d p_j = \sum_{j=1}^d q_j = 1$. This divergence can be used with general vectors (i.e. not necessarily probability distributions) and then we get the Generalised I-divergence, $d_F(\mathbf{p}, \mathbf{q}) = \sum_{j=1}^d p_j \log \frac{p_j}{q_j} - \sum_{j=1}^d (p_j - q_j)$. Other divergences include the Itakura-Saito divergence, the Mahalanobis distance and the logistic loss, corresponding to $F(x) = \log(x)$, $F(\mathbf{x}) = \mathbf{x}\Sigma^{-1}\mathbf{x}$, with Σ the data covariance matrix, and $F(x) = x \log x + (1 - x) \log(1 - x)$ respectively.

2.1 Properties of Bregman Divergences

First note that, in general, $d_F(\mathbf{p}, \mathbf{q}) \neq d_F(\mathbf{q}, \mathbf{p})$. However we can create symmetric divergences:

$$\begin{aligned}
 S_F(\mathbf{p}, \mathbf{q}) &= \frac{1}{2}(d_F(\mathbf{p}, \mathbf{q}) + d_F(\mathbf{q}, \mathbf{p})) \\
 &= \frac{1}{2}\langle \mathbf{p} - \mathbf{q}, \nabla F(\mathbf{p}) - \nabla F(\mathbf{q}) \rangle
 \end{aligned}$$

This gives us a divergence measured on the space S and its derivative space ∇S . All Bregman divergences satisfy

- Non-negativity** $d_F(\mathbf{p}, \mathbf{q}) \geq 0$ with equality if and only if $\mathbf{p} = \mathbf{q}$.
- Convexity** but only guaranteed in the first parameter.
- Linearity** $d_{aF_1+bF_2}(\mathbf{p}, \mathbf{q}) = ad_{F_1}(\mathbf{p}, \mathbf{q}) + bd_{F_2}(\mathbf{p}, \mathbf{q})$

A fuller description of the properties of Bregman divergences can be found in [2]. However, this leaves open the question as to which Bregman divergence is the best one to use for any particular data set, something which will inevitably depend on the distribution of the data set. To find an answer to this, we digress to re-state the properties of the exponential family of distributions.

3 The Exponential Family

[2] have shown that there is bijection between a set of Bregman divergences and members of the regular exponential family of probability distributions. The exponential family of distributions is a surprisingly wide family whose members have distributions of the form

$$p_{G,\theta}(\mathbf{x}) = \exp(\langle \mathbf{t}(\mathbf{x}), \theta \rangle - G(\theta))p_0(\mathbf{t}(\mathbf{x})) \tag{2}$$

where $\mathbf{t}(\mathbf{x})$ is known as the natural statistic, θ is known as the natural parameter and $G(\theta)$ is the cumulant function which defines the exponential family. An example of the exponential family are

The 1 dimensional Gaussian with unit variance

$$\begin{aligned}
 p_{G,\theta}(x) &= \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{x-\mu}{2})^2} \\
 &= \frac{e^{-x^2}}{\sqrt{2\pi}} e^{x\mu - \frac{\mu^2}{2}}
 \end{aligned}$$

So that $t(x) = x$

$$\theta = \mu$$

$$\text{and } G(\theta) = \frac{\theta^2}{2} = \frac{\mu^2}{2}$$

Other well known members of this family are the bernoulli, multinomial, beta, Dirichlet, Poisson, Laplace, gamma and Rayleigh distributions. In the remainder of this paper, we consider only regular exponential families in which $\mathbf{t}(\mathbf{x}) = \mathbf{x}$.

We define the expectation of X with respect to $p_{G,\theta}$ to be

$$\mu = E_{p_{G,\theta}}[X] = \int_{\mathbb{R}^d} \mathbf{x} p_{G,\theta}(\mathbf{x}) d\mathbf{x} \tag{3}$$

It can be shown [2] that there is a bijection between the set of expected values, μ , and the set of natural parameters, θ . In fact, let d_F be the Bregman divergence corresponding to the distribution, $p_{G,\theta}$. Then let $g(\cdot) = \nabla G$ and let $f = \nabla F$. Then $\mu = g(\theta)$ and $\theta = f(\mu)$, which is readily verified for the distributions above.

Consider a member of the regular exponential family with known cumulant function, $G(\theta)$. Then $G(\cdot)$ is a closed convex function. Define its conjugate function as

$$F(\mathbf{x}) = \sup_{\theta} \{\langle \mathbf{x}, \theta \rangle - G(\theta)\} \tag{4}$$

Then there is an unique θ^* which attains the supremum and $F(\cdot)$ is also a convex function. If the domain of F is S and the domain of G is Θ , then (S, F) is the Legendre dual of (Θ, G) . In particular, there exists a θ such that $F(\mu) = \langle \mu, \theta \rangle - G(\theta)$. Differentiating and setting the derivative to 0, we see that $g(\theta) = \mu$ and $f(\mu) = \theta$; then since $G(\cdot)$ is strictly convex, $F(\cdot)$ is too and so can be used to define a Bregman divergence. Consider two members of an exponential family with natural parameters, θ_1 and θ_2 , and expectations, μ_1 and μ_2 . Then it can be shown that minimising the Bregman divergence with respect to the cumulant function between the natural parameters is equivalent to minimising the Bregman divergence with respect to the dual function (but in the opposite direction) between the expectations. Also it can be shown that maximising the likelihood of a data set is equivalent to minimising the associated Bregman divergence between the mean of the distribution and the data.

In practical terms, we might fit a particular member of the exponential family to a data set which means we have determined the cumulant function, $G(\cdot)$. We then identify the dual function, $F(\cdot)$, based on which we can find the Bregman divergence $d_F(\cdot)$ knowing that minimising the Bregman divergence between the mean of the distribution and its natural statistics maximises the log likelihood of the distribution under this probability density function.

4 Topology Preserving Mappings

2 use Bregman divergences in the K Means algorithm to create a family of clustering algorithms. The centres are initialised to K data points within the domain of the data and, as with K Means, they iterate between

1. The assignment step: every data point is assigned to the cluster whose centre has minimum Bregman divergence from it.
2. The re-estimation step: each prototype is estimated to lie at the mean of those data points allocated to its cluster.

We note that there are potentially two Bregman divergences that can be used: for data point \mathbf{x}_i and centre μ we could use either $d_F(\mathbf{x}_i, \mu)$ or $d_F(\mu, \mathbf{x}_i)$. Of course these are equivalent for Euclidean distances (*the symmetric Bregman divergence*) but have somewhat different properties with the non-symmetric divergences. The latter definition of divergences is used for the results we show later.

A topology preserving mapping of a data set is a mapping which retains some property of the data set in an ordered manner. For example, in the visual cortex, we have neurons which have optimal response to different orientation of bars. Crucially, however, as we traverse part of the cortex, the optimal orientation changes smoothly and gradually: nearby neurons respond optimally to similar orientations. Topographic mappings are rather ubiquitous in the cortex, appearing for example in the visual, auditory, somatosensory and motor cortex.

4.1 SOM Based Method

Kohonen's algorithm 4 is exceedingly simple - the network is a simple 2-layer network and competition takes place between the output neurons; however not only are the weights into the winning neuron updated but also the weights into its neighbours. Kohonen defined a neighbourhood function $f(i, i^*)$ of the winning neuron i^* . The neighbourhood function is a function of the distance between i and i^* . A typical function is the Difference of Gaussians function; thus if unit i is at point \mathbf{r}_i in the output layer then

$$f(i, i^*) = a \exp\left(\frac{-|r_i - r_{i^*}|^2}{2\sigma^2}\right) - b \exp\left(\frac{-|r_i - r_{i^*}|^2}{2\sigma_1^2}\right) \quad (5)$$

where r_k is the position in neuron space of the k^{th} centre: if the neuron space is 1 dimensional, $r_k = k$ is a typical choice; if the neuron space is 2 dimensional, $r_k = (x_k, y_k)$, its two dimensional Cartesian coordinates.

4.2 Simulations

We wish to classify magnetic resonance images (MRI) of the brain into the various classes of tissue, grey matter, white matter and so on. We train a SOM

with a data set composed of 5 pixels by 5 pixels window from a particular image which is 179 pixels by 179 pixels. When the SOM trained, we need some method to display the results: we wish to keep the two dimensional representation of the physical brain so that nearby pixels are actually from the same type of matter and any human viewer can easily verify that the different regions are being located according to different structure in the brain.

Let \mathbf{x}_i be the i^{th} sample in our data set. Then \mathbf{x}_i is 25 dimensional. Then we calculate the responsibility that each of the 100 neurons in our mapping has for \mathbf{x}_i using

$$r_{ki} = \frac{\exp(-\gamma d_{ki})}{\sum_{j=1}^{100} \exp(-\gamma d_{ji})} \tag{6}$$

where d_{ki} is the divergence between the k^{th} neuron's centre in data space and the i^{th} data sample.

We use these responsibilities to calculate a position for the i^{th} data sample in the two dimensional neuron space using

$$\mathbf{y}_i = \sum_{j=1}^{100} r_{ji} \mathbf{t}_j \tag{7}$$

where \mathbf{t}_j is the two dimensional coordinate of the j^{th} neuron in neuron space.

We create a two dimensional SOM based on a variety of divergences. In order to display the results we show a slice of the brain image in Figure 2 in which we take one dimension of a two dimensional SOM and associate this with the red channel of an RGB image while the other dimension of the SOM is associated with the blue channel i.e. $y_{i,1}$ defines the red part of the image while $y_{i,2}$ defines the blue part. We see that we are able to capture the main ventricles in the Euclidean SOM but we capture more detail in the SOM which uses the Itakura-Saito divergence.

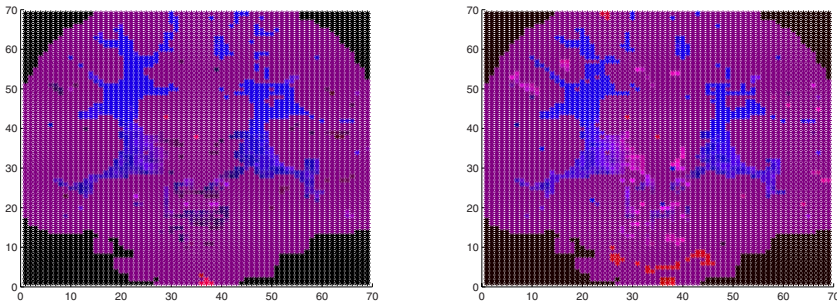


Fig. 2. Left: SOM using Euclidean distances. Right: SOM using I-S. divergences. The red colour shows its position on the horizontal axis while the blue shows its position on the vertical axis.

5 Conclusion

We have reviewed Bregman divergences and shown the very close relationship between these divergences and the exponential family of probability density functions: if we know the pdf of a data set, we can choose the optimal divergence associated with that dataset.

In particular we have applied Bregman divergences the self-organising map and shown that different divergences may reveal more structure than the Euclidean distances. We have not shown all here but different divergences do reveal different types of information and we believe that a data analyst would be well advised to investigate all of these divergences when dealing with a new dataset.

Acknowledgement. This research was supported by the Ministry of Knowledge Economy, Korea, Under the ITFSIP(IT Foreign Specialist Inviting Program) supervised by the IITA(Institute of Information Technology Advancement).

References

1. Azoury, K.S., Warmouth, M.K.: Relative loss bounds for on-line density estimation with the exponential family of distributions. *Machine Learning* (43), 211–246 (2001)
2. Banerjee, A., Meruga, S., Dhillon, I., Ghosh, J.: Clustering with bregman divergences. *Journal of Machine Learning Research* 6, 1705–1749 (2005)
3. Collins, M., Dasgupta, S., Shapire, R.E.: A generalization of principal component analysis to the exponential family. In: *Nips14* (2002)
4. Kohonen, T.: *Self-Organising Maps*. Springer, Heidelberg (1995)
5. Lee, J.A., Verleysen, M.: *Nonlinear Dimensionality Reduction*. Springer, Heidelberg (2007)
6. Neilsen, F., Boissonnat, J.-D., Nock, R.: Bregman voronoi diagrams: Properties, algorithms and applications (submitted, 2007)
7. Neilsen, F., Boissonnat, J.-D., Nock, R.: On bregman voronoi diagrams. In: *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pp. 746–755 (2007)

Feature Locations in Images

Hokun Kim¹, Colin Fyfe², and Hanseok Ko¹

¹ Intelligent Signal Processing Lab,
Korea University, Korea
{hkkim,hsko}@ispl.korea.ac.kr

² School of Computing,
University of the West of Scotland, UK
colin.fyfe@uws.ac.uk

Abstract. We review the recent technique of two dimensional canonical correlation analysis and illustrate its use as a method for identification of the location of particular features in a data set.

1 Canonical Correlation Analysis

Canonical Correlation Analysis [6] is used when we have two data sets which we believe have some underlying correlation. Consider two sets of input data, from which we draw iid samples to form a pair of input vectors, \mathbf{x}_1 and \mathbf{x}_2 . Then in classical CCA, we attempt to find the linear combination of the variables which gives us maximum correlation between the combinations. Let

$$y_1 = \mathbf{w}_1 \mathbf{x}_1 = \sum_j w_{1j} x_{1j} \quad (1)$$

$$y_2 = \mathbf{w}_2 \mathbf{x}_2 = \sum_j w_{2j} x_{2j} \quad (2)$$

Then we wish to find those values of \mathbf{w}_1 and \mathbf{w}_2 which maximise the correlation between y_1 and y_2 . Whereas Principal Components Analysis and Factor Analysis deals with the interrelationships within a set of variables, CCA deals with the relationships between two sets of variables. If the relation between y_1 and y_2 is believed to be causal, we may view the process as one of finding the best predictor of the set \mathbf{x}_2 by the set \mathbf{x}_1 and similarly of finding the most predictable criterion in the set \mathbf{x}_2 from the \mathbf{x}_1 data set.

CCA was one of the first methods used by image processing scientists for the task of image registration. Variants based on kernels [4], artificial neural networks [3,2] and probabilistic models [1] have also been proposed. In this paper, we utilise a new variant which attempts to find correlations while maintaining the two dimensional structure of an image.

2 2DCCA

The standard methods of image processing converts the $m \times n$ image into a vector of length $mn \times 1$ which is then used for subsequent processing. Since

this often involves matrices like the $mn \times mn$ covariance matrix, the subsequent processing can be very computationally demanding. Recently, a new approach to image identification has been taken [7] in which the planar aspect of an image is retained. Let an image be defined as $A_i \in \mathfrak{R}^{m \times n}$, an $m \times n$ image. Let $X \in \mathfrak{R}^{n \times d}$, $d \leq n$, have orthonormal columns so that $X^T X = I$, the d -dimensional identity matrix. Then we may ‘project’ A onto X to get $Y_i = A_i X$ with a view to retaining as much of the variance in the data set as possible. An algorithm was developed to do this by finding the matrix,

$$G = \frac{1}{N} \sum_{i=1}^N (A_i - \bar{A})^T (A_i - \bar{A}) \tag{3}$$

It was shown that the optimal value of the projection matrix X is given by the eigenvectors, X_1, \dots, X_d of G associated with the largest eigenvalues. Note that G is $n \times n$ and so a much less computationally intensive problem than the standard method which would have a covariance matrix of size $m \times n \times m \times n$. A two sided version of this technique was developed

$$C_i = Z A_i X \tag{4}$$

which was used for face identification in [8].

Very recently [9], a similar approach has been taken to Canonical Correlation Analysis: let us have two sets of image data denoted by $A_i \in \mathfrak{R}^{m_1 \times n_1}$ and $B_i \in \mathfrak{R}^{m_2 \times n_2}$, $i = 1, \dots, N$ in which the data samples are related at any one time. Assume the data is centered or carry out a centering operation, $A_i \leftarrow (A_i - \bar{A})$. We will be interested in finding only the first two sided canonical correlations but the extension to more than the first canonical correlation is obvious. We wish to find left transforms l_A and l_B and right transforms r_A and r_B which maximise the correlation between $l_A^T A r_A$ and $l_B^T B r_B$ under the constraints that variance of $l_A^T A r_A$ and $l_B^T B r_B$ are both 1. We define

$$\begin{aligned} \Sigma_{AB}^r &= \frac{1}{N} \sum_{i=1}^N A_i r_A r_B^T B_i^T \\ \Sigma_{AA}^r &= \frac{1}{N} \sum_{i=1}^N A_i r_A r_A^T A_i^T \\ \Sigma_{BB}^r &= \frac{1}{N} \sum_{i=1}^N B_i r_B r_B^T B_i^T \end{aligned}$$

Then the covariance

$$cov(l_A^T A r_A, l_B^T B r_B) = l_A^T \Sigma_{AB}^r l_B \tag{5}$$

and the constraints are also expressible with these ‘covariance’ matrices. Note that we can also express the covariance matrix as

$$cov(l_A^T A r_A, l_B^T B r_B) = r_A^T \Sigma_{AB}^l r_B \tag{6}$$

where we have defined the matrix $\Sigma_{AB}^l = \frac{1}{N} \sum_{i=1}^N A_i^T l_A l_B^T B_i$. This suggests an obvious optimization algorithm: use the current estimates of r_A and r_B and solve the generalised eigenproblem,

$$\begin{bmatrix} 0 & \Sigma_{AB}^r \\ \Sigma_{BA}^r & 0 \end{bmatrix} \begin{bmatrix} l_A \\ l_B \end{bmatrix} = \lambda \begin{bmatrix} \Sigma_{AA}^r & 0 \\ 0 & \Sigma_{BB}^r \end{bmatrix} \begin{bmatrix} l_A \\ l_B \end{bmatrix} \tag{7}$$

Then use the current estimates of l_A and l_B on the eigenproblem,

$$\begin{bmatrix} 0 & \Sigma_{AB}^l \\ \Sigma_{BA}^l & 0 \end{bmatrix} \begin{bmatrix} r_A \\ r_B \end{bmatrix} = \lambda \begin{bmatrix} \Sigma_{AA}^l & 0 \\ 0 & \Sigma_{BB}^l \end{bmatrix} \begin{bmatrix} r_A \\ r_B \end{bmatrix} \tag{8}$$

and repeat these two stages till convergence. We have found that convergence is typically very fast and in the simulations herein, we used only three iterations.

3 Simulations

3.1 Artificial Data

We begin with an easy data set: we create a pseudo-image by creating a random 100*100 matrix of white noise drawn from a uniform distribution in [0,0.1] and create a signal by embedding a 5*5 square of amplitude 1 in this image. Let the signal be at position (j, k) in the original image; then we create our A image set by sampling the signal and its surrounds so that

$$A_{3i+m+1} = \begin{bmatrix} a_{j+i,k+m} & a_{j+i,k+m+1} & a_{j+i,k+m+2} & a_{j+i,k+m+3} & a_{j+i,k+m+4} \\ a_{j+i+1,k+m} & & \dots & & a_{j+i+1,k+m+4} \\ a_{j+i+2,k+m} & & \dots & & a_{j+i+2,k+m+4} \\ a_{j+i+3,k+m} & & \dots & & a_{j+i+3,k+m+4} \\ a_{j+i+4,k+m} & & \dots & & a_{j+i+4,k+m+4} \end{bmatrix} \tag{9}$$

where i, m grow from 0 to 2 which gives us 9 images in our A data set. We then create all possible B data sets of 9 images each in a similar manner and can

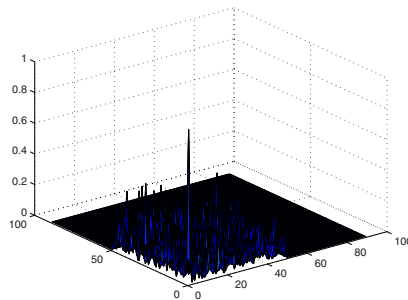


Fig. 1. The position of the signal is clearly identifiable



Fig. 2. The source image, “Ruby”



Fig. 3. The target image, “Colin”

clearly identify the greatest correlation. The resulting values of the correlations are shown in Figure 1; we can clearly identify the position of the signal since the correlation between the A data and the B data is much larger at this point since it is 1 while every other correlation is less than 0.2.

We similarly carried out an experiment where our target was less pronounced i.e. it was also random data. Again a correlation of 1 identified the exact position in the image from which the A dataset was drawn. However this time there were sections of the B image which gave correlations close to (but not equal to) 1.

3.2 Real Data

The image “Ruby” is shown in Figure 2. It is 193*200 pixels and was first converted from RGB format to YCrCb and only the Y component was used in

subsequent processing. We selected a 10×10 section around Ruby's eye and used this in the same way as above for our A dataset: because we had 100 pixels per slice this time we used 5×5 starting positions (for each 10×10 image, A_i). Our first experiment was to ensure that we could reliably identify in the image the position of the A data set; again the correlation of 1 was easily found though, as with our random data, other correlations of close to 1 were also found. Nevertheless the method easily identified the position of Ruby's eye.

We then used this mask, the A data set from Ruby's eye with a B data set from a different image, "Colin" (see Figure 3) which is 224×202 pixels. We see that Colin is somewhat different from Ruby but nevertheless Colin's right eye was identifiable from the correlations with Ruby's.

4 Conclusion

We have reviewed the method of 2D Canonical Correlation Analysis and used it for identification of similar features in different images. Our long term goal is rather more ambitious: we will investigate whether the method can be used for image registration - the identification of the same points in more than one image. The results from the last experiment which identified Colin's eye from a mask of Ruby's eye gives us confidence that we will be successful.

Acknowledgement. This research was supported by the Ministry of Knowledge Economy, Korea, Under the ITFSIP (IT Foreign Specialist Inviting Program) supervised by the IITA (Institute of Information Technology Advancement).

References

1. Fyfe, C., Leen, G., Lai, P.L.: Gaussian processes for canonical correlation analysis. *Neurocomputing* (2008)
2. Gou, Z.K., Fyfe, C.: A canonical correlation neural network for multicollinearity and functional data. *Neural Networks* (2003)
3. Lai, P.L., Fyfe, C.: A neural network implementation of canonical correlation analysis. *Neural Networks* 12(10), 1391–1397 (1999)
4. Lai, P.L., Fyfe, C.: Kernel and nonlinear canonical correlation analysis. *International Journal of Neural Systems* 10(5), 365–377 (2001)
5. Lee, S.H., Choi, S.: Two-dimensional canonical correlation analysis. *IEEE Signal Processing Letters* 14(10), 735–738 (2007)
6. Mardia, K.V., Kent, J.T., Bibby, J.M.: *Multivariate Analysis*. Academic Press, London (1979)
7. Yang, J., Zhang, D., Frangi, A.F., Yang, J.Y.: Two dimensional pca: a new approach to appearance-based face representation and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26(1), 131–137 (2004)
8. Zhang, D., Zhou, Z.H.: $(2d)^2$ pca: 2-directional 2-dimensional pca for efficient face representation and recognition. *some journal* 26(1), 131–137 (2004)

A Hierarchical Self-organised Classification of ‘Multinational’ Corporations

Khurshid Ahmad¹, Chaoxin Zheng¹, and Colm Kearney²

¹ Department of Computer Science, Trinity College, Dublin

² School of Business Studies, Trinity College, Dublin

kahmad@cs.tcd.ie, chaoxin.zheng@cs.tcd.ie, colm.kearney@tcd.ie

Abstract. Classification of entities, for example, into national states, into social groups, into business enterprises and into scientific taxa, is an enduring problem in neural computing. In this paper, we look at the problems faced by researchers in developing a taxonomy of ‘multinationality’ and explore the use of hierarchical SOMs in ‘discovering’ a taxonomy of multinational corporations (MNCs).

Keywords: Hierarchical Self Organising Maps, Multinational Corporations.

1 Introduction

Financial engineering has seen a number of applications of neural networks. These systems have been used: (a) for classifying of investment opportunities (Yu, Wang and Lai 2008); (b) in predicting the behaviour of markets (McNeils 2005, Taskaya-Temezil, Casey and Ahmad 2005); and (c) in categorizing large volumes of financial texts (Manomaisupat, Vrusias & Ahmad 2006).

In this paper we will look at the role of neural networks in the classification of entities discussed in the literature on the theory of the firm where problems of taxonomic organization are at the forefront. We look at the classification of ‘multinational’ corporations (MNCs) based on their multifaceted attributes. The compilation of the attributes requires acquisition and metrication of data relating to a corporation’s ‘engagement with and exposure to’ domestic, regional, trans-regional and global markets (Aggarwal, Berril and Kearney, forthcoming).

2 Motivation: Towards a Taxonomy of Multinational Corporations

The classification of corporations is an important issue in business and finance both in terms of the conceptual basis of such a classification and in terms of the practical import of the evolution, sustenance and obsolescence of this critical component of economies large and small. Scholars of finance in general, and international business and international marketing in particular, have stressed the need for a taxonomic classification of multinational companies (see, for example, Barry and Kearney 2006).

The multinational corporation has a number of facets: An MNC conducts research, is involved in marketing and sales, and invests across geographical and political boundaries. An MNC maintains a 'main' or 'head' office in one particular location, it sends its key managers to parts of the world that have different political and cultural values and systems. Researchers in international business are interested in quantifying: (a) what happens in the offices of an MNC 'abroad' by using *performance* attributes of an MNC; (b) what financial and physical resources are deployed away from 'home', by using the *structural* attributes; and, (c) how well an MNC is or will be performing by using *attitudinal* attributes. The reliance on one single attribute, for example *foreign sales as a percentage of total sales*, to measure the degree of internationalisation of an MNC, makes such a measurement very vulnerable to the inevitable errors in the values of the single attribute. Nevertheless, a review of over 100 papers, published between 1970 to 2006, on the topic of classifying MNCs, shows that just about half the researchers have used a single variable, whilst the other half conducted multivariable studies (Aggarwal, Berrill and Kearney, *forthcoming*). We intend to find out whether the use of self-organising maps has a role in classifying MNCs.

3 A Note on Hierarchical SOMs

A self-organising map (SOM) is used to map high-dimensional observations onto a lower dimensional map whilst preserving the topological relations between the represented objects with a degree of fidelity. The SOM algorithm is based on a winner-takes-all-strategy where individual nodes in the low-dimensional map 'win' the right to represent objects in the high dimensional space. The SOM seldom produces discernible clusters and it is important to analyse the output of an SOM using other clustering techniques to ascertain the presence or absence of clusters in the output (see, for example Ahmad, Vrusias and Ledford 2001).

The hierarchical SOM (HSOM), proposed originally by, amongst others, Lampinen and Oja (1992), comprises n independent SOM's organised in layers. The first layer is trained using an input feature vector. Once trained, the winning nodes in the first map act as input to the next layer and so on. The hierarchical map then helps in visualizing the input data at different levels of taxonomic organization. The top layer may be used to identify the presence of clusters and the lower levels representing sub-clusters. It has been argued that an HSOM can be trained more quickly than a single SOM and that HSOM's can be used to map data sets with higher dimensions than is effectively possible with a single SOM (see, Vicente & Vellido 2008 for an elaborate review on the subject.). An algorithm for training an HSOM is given below.

```
Train an m-layer HSOM using the input data set
comprising  $n$ -vectors  $[I_1, I_2, I_3, \dots, I_n]$ ;
```

```
For  $c = \text{cycle } 1$  to cycle  $C$ 
```

```
  For  $t = \text{input } 1$  to input  $n$ 
```

```
    Present  $I_t$  to the network;
```

```
    For  $i = \text{layer } 1$  to layer  $m$ 
```

Find the winner $w_{t,i}$ with the smallest distance to the input from layer $i-1$ using the criterion below:

$$w_{t,i} = \arg \min_j \{ \|w_{j,i} - w_{t,i-1}\| \}$$

Update the weights of nodes around $w_{t,i}$ at layer i using

$$w_{t+1,i} = w_{t,i} + \alpha_{t,i} h_{t,i} [I_t - w_{t,i}]$$

Present $w_{t,i}$ as input to the layer $i+1$;

Update the learning rate and neighborhood size h at each layer using:

$$\alpha_{t,i} = \alpha^{start} \exp\left(\frac{-(t + c \times n) \ln(\alpha^{end} / \alpha^{start})}{C \times n}\right)$$

$$h_{t,i} = h^{start} \exp\left(\frac{-(t + c \times n) \ln(h^{end} / h^{start})}{C \times n}\right)$$

where the α^{start} (h^{start}) & α^{end} (h^{end}) are the final and initial value of learning rate and neighborhood size;

End i

End t

End c

The HSOM used in this paper for the classification of MNCs has three layers comprising 10x10, 5x5, and 2x2 neurons respectively. The initial neighborhood size for all the three maps is a quarter of the size of the map, decreasing exponentially to finally 1. The initial learning rates for all the three layers are all set to 0.9, which is also exponentially decreased to 0.1.

During training, the feature vectors (representing each of the corporations) are presented to the HSOM one at a time and the training takes over a predetermined number of cycles – 10 is the number we have used. During each cycle, the order of presentation of the feature vectors is randomly selected. For testing, a proportion of company data is randomly selected and then used only for testing purposes.

4 Towards a Multivariate Classification of MNCs

We will now present two multivariate studies for classifying corporations. The classic study of Sullivan (1994) is repeated using a 3 layer HSOM; we use a data set of 25 corporations, each comprising 5 attributes. This is followed by analyzing a larger and more modern data set created by Aggarwal, Birrel and Kearney. The data set used in this paper contains 100 MNCs with 7 attributes per corporation. The case studies are multivariate as in both we look at the variance in the value of more than one attribute.

4.1 Case Study I: Multivariate Analysis and Sullivan’s Classification of MNC

Sullivan (1994) has suggested a multivariable study of the measures to compute the degree of internationalization of an MNC.

Table 1. The nine key variables used in Sullivan (1994) divided into three key attributes

Performance	Foreign Sales/ Total Sales (FSTS), Export Sales/Total Sales (ESTS), Foreign Profits/Total Profits (FPTP), R&D Intensity (RDI), Advertising Intensity Performance (AIP)
Structure	Foreign Assets/Total Assets (FATA), Overseas Subsidiaries/Total Subsidiaries (OSTS)
Attitudinal	Top Managements’ International Experience (TMIE), Psychic Dispersion of International Operations (PDIO)

The author has collated data on the nine attributes and performed *item analysis* (see Field 2005) on these attributes for 74 corporations (see Table 1). This analysis helped in finding that the key attributes, FSTS, FATA, OSTs, TMIE and PDIO, show statistically significant correlation with each other. These results were confirmed by *factor analysis*. A weighted linear combination of the five attributes was termed *degree of internationalization* (DOI) and it was found to be distributed normally.

Sullivan (1994: 336) has provided DOI-rankings of 25 companies based on the estimates of the five given attributes and one computed attribute (DOI). We have used this data set to train a hierarchical SOM to explore what kind of clustering we obtain by simply using the rank order of the 5 attributes for the 25 companies. The training set for our HSOM comprised 22 feature vectors each with the rank-order information about the company’s FATA, FSTS, OSTs, PDIO and TMIE. The HSOM itself comprises of a root network of 10x10 nodes. This network, which receives the feature vectors, acts as input to the next 5x5 network in the hierarchy. Finally, the 5x5 network acts as input to a 2x2 network. Recall that our system selects about 10% of the vectors randomly for testing – in the case of Sullivan’s 25 corporations, 3 were chosen randomly for testing – *American Cyanamid*, *Avon*, and *Pfizer*.

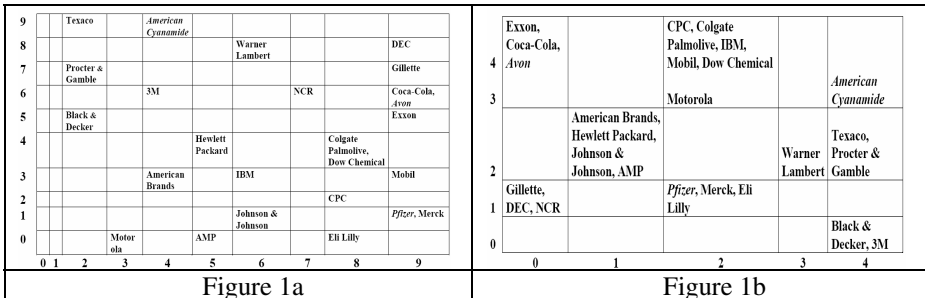


Fig. 1. HSOM Classification - The mapping of Sullivan’s 25 companies. Figure 1a shows the initial root map (10x10) and the second level (5x5) is shown in Figure 1b.

The initial mapping from the 5 dimensional vector onto the 2-dimensional 10x10 surface shows a scatter of the corporations with no significant clusters (Figure 1a). The weights of the winning node in the 10x10 network are then used to train the 5x5 network (Figure 1b). This network shows an emergence of some structure with the corporations with highest ranks in FATA and FSTS clustering together. Finally, the weights of the winning nodes in the 5x5 network are used in the training of the final 2x2 map and there is evidence of clustering here.

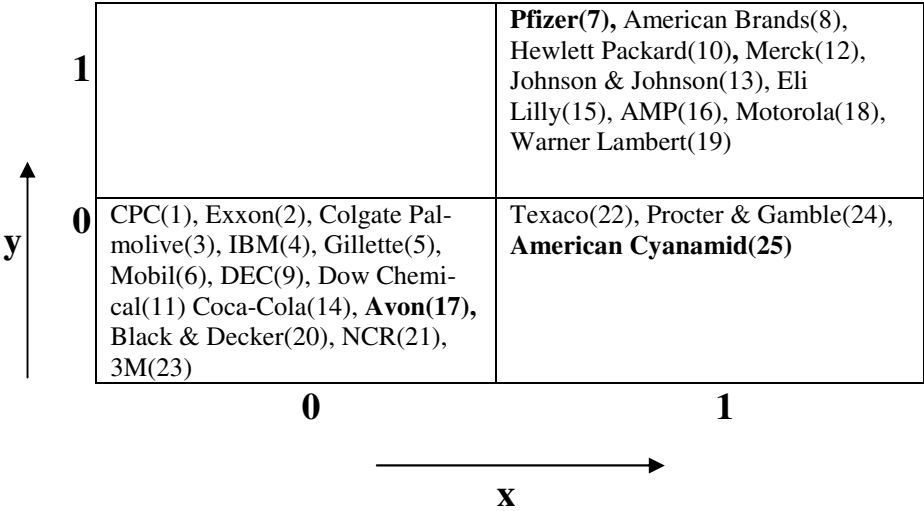


Fig. 2. The final level in the HSOM Classification for MNC data in Sullivan (Numbers in parentheses are the DOI rankings by Sullivan, but were not used in training.) The values for FSTS, FATA, OSTS, TMIE and PDIO, were used to train the HSOM. Note that the data related to 3 companies (Avon, American Cyanamid and Pfizer) was only used for testing.

The 2x2 map can be divided into four quadrants that we count clockwise through the x-y co-ordinates 00, 01, 11 and 10 as 1st, 2nd, 3rd and 4th quadrants. In the 1st quadrant (00, Figure 2), 8 out of the 11 top ranking corporations have been classified together; there are 4 lowest ranked organisations also in this quadrant. The 2nd quadrant (01) is empty. The 3rd quadrant (11) has all intermediate ranking corporations, from 7th to 19th. The 4th quadrant (10) has lowest ranked corporations. Test feature vectors (for the corporations *Avon* (rank 17), *Pfizer*(7) and *American Cyanamid* (rank 25)) show that whilst the lowest ranked was classified correctly, the ambivalence of the first and third quarter showed in the assignment of highly ranked corporation (*Pfizer*) in the intermediate rank cluster, and vice versa for *Avon*. With the external knowledge of industry type, one can argue that the HSOM has correctly placed *Pfizer* with other pharmaceutical corporations (*Eli-Lilly*, *Johnson and Johnson* and *Warner Lambert*). The key here is to see whether one or more attributes dominate Sullivan’s DOI-rank order. We have computed the average rank order for each of the variables in the three quadrants. It appears that the average belongingness to the clusters is decided by two variables: *FSTS* and *FATA*.

It has been argued that an index like that of Sullivan’s obscures the fact that one may get the same value of *Internationalization* for different values of *Performance, Structure and Attitude* leading to different kinds of multi-nationality (see, for instance, Ramaswamy, Kroeck and Renforth 1996).

We now turn to our ongoing studies of the multinational corporations where we will look at data for the Fortune 500 list of corporations.

4.2 Case Study II: An HSOM Analysis of Fortune 100 Corporations

Following a study of over 350 corporations listed in Fortune 500 list of ‘top’ corporations, Aggarwal, Birrel and Kearney have devised a new MNC classification scheme. The attributes of these corporations were established by looking at how taxa are developed in agriculture, biology, chemistry, education, IT, Information Sciences, psychology and religion. Performance, structural and behavioural attributes were then outlined by Aggarwal et al (Table 2); there are similarities and differences with that of Sullivan’s (Table 1).

Table 2. Classification attributes of corporations listed by Aggarwal et al

Attributes	Values
Performance	Foreign Sales; Subsidiaries; Foreign Assets; Foreign Income; Foreign Joint Ventures; Mergers & Acquisitions; International Transactions.
Structural	Global Accounts; Industry Details; Foreign Employees; Foreign Equity; Foreign Exchange Listing.
Behavioural	World Mandates; Patents; Research & Development; Foreign Taxation; Unclear

The authors have re-interpreted the notion of *multinationality* operations performed by corporations either within or across countries and regions. They have created a 2x4 matrix, where rows label depth of engagement (through trade and investment), and columns label the breadth of the geographical spread (comprising domestic, regional, trans-regional and global breadths). This results in firms classified on a spectrum from purely domestic corporations to the deeply global ones. The authors suggest a 9-point scale in this respect over the space of 16 corporation types. Furthermore, Aggarwal et al’s taxonomy distinguishes between publicly quoted corporations and state owned ones, and has used an industry classification – including *finance & insurance, information technology, manufacturing, retail, wholesale, and utilities*. In their data set the authors have observed the uneven distribution of the 351 corporations across the 9-point multinationality spectrum: there are 171 corporations that trade and invest on a trans-regional basis, 89 have global investments and trade trans-regionally, 13 that do both globally, and there 27 exclusively domestic corporations. The location, age, and size are used as attributes in conjunction with others described above for ‘computing’ multinationality. Sullivan’s attitudinal attributes are not used.

We trained an HSOM (10x10, 5X5 and 2x2 layers) with the data for the top 90 corporations listed in Fortune 500 and then tested the HSOM with 10 corporations. The training and testing samples were randomly selected by our program. The 10x10

Table 3. Classification of corporations in the 2x2 top layer of an HSOM

Multinationality: Sales				Multinationality: Subsidiaries			
Geog. Spread	2X2 Map Coords.			Geog. Spread	2X2 Map Coords		
	00	01	11		00	01	11
Global	0	7	3	Global	12	16	8
Transnational	24	36	16	Transnational	14	27	12
Regional	0	2	1	Regional	0	2	0
Domestic	2	9	0	Domestic	0	9	0
TOTAL	26	54	20	TOTAL	26	54	20

The corporations with domestic and regional subsidiaries only have been clustered in the 2nd quadrant as well.

5 Afterword

We have attempted to demonstrate the utility of self-organising maps in general, and the hierarchical SOM's in particular, in a subject domain where the definition of an entity, the multi-national corporation, eludes the scholars. The hierarchical SOM generates not only a map, at the root SOM (10x10 in our case), but also what Vicente and Vellido (2008) have called an *atlas* at the top level (the 2x2 layer). We have seen the key attributes that were forcing the cluster formation in the case studies. The use of multinationality data, quantified through the use of sales and subsidiaries, requires further analysis, especially since our data set is dominated by trans-regional corporations.

The classification found in the 2x2 map, and indeed the other larger 5x5 and 10x10 maps, suggest that the two axes of the map may represent two conceptual variables that could be used to classify corporations, multinational or otherwise.

There is much room for improvement in our data sets. We are in the process of collating the data for all FORTUNE 500 corporations, but this effort is exacerbated by the fact that the data provided in the FORTUNE list has to be supplemented from the documents of each corporation individually for establishing, for example the geographical spread of sales and subsidiaries.

In terms of neural computing, the usage of growing HSOMs appears very attractive, especially when corporations with novel attributes come in the market place.

Acknowledgements. The authors wish to thank Donal Holland for helping with the layout and proof reading of the paper. Thanks are due to Maria F. O'Connor for help with the running of the self-organising maps.

References

1. Aggarwal, R., Berrill, J., Kearney, C.: A taxonomy of Multinationality: Towards a Classification System for MNCs (in preparation)
2. Ahmad, K., Vrusias, B., Ledford, A.: Choosing Feature Sets for Training and Testing Self-Organising Maps: A Case Study. *Neural Computing & Applications* 10, 56–66 (2001)

3. Barry, F., Kearney, C.: MNEs and Industrial structure in Host Countries: A Portfolio Analysis of Irish Manufacturing. *Journal of International Business Studies* 37(3), 392–406 (2006)
4. Field, A.P.: *Discovering Statistics using SPSS*, 2nd edn., ch. 15. Sage, London
5. Lampinen, J., Oja, E.: Clustering properties of hierarchical self-organizing maps. *Journal of Mathematical Imaging and Vision* 2, 261–272 (1992)
6. Manomaisupat, P., Vrusias, B., Ahmad, K.: Categorization of Large Text Collections: Feature selection for unsupervised and supervised neural networks. In: Corchado, E., Yin, H., Botti, V., Fyfe, C. (eds.) *IDEAL 2006. LNCS*, vol. 4224, pp. 1003–1013. Springer, Heidelberg (2006)
7. McNeils, P.D.: *Neural Networks in Finance – Gaining Predictive Edge in the Market*. Elsevier, Amsterdam (2005)
8. Popoola, A., Ahmad, K.: Testing the Suitability of Wavelet Pre-processing for TSK Fuzzy Models. In: *Proc. FUZZ-IEEE 2006: Int. Conference on Fuzzy Systems Networks*, pp. 6655–6659 (2006)
9. Ramaswamy, K., Kroeck, K., Renforth, W.: Measuring the Degree of Internationalisation of a Firm: A Comment. *Journal of International Business Studies* 27(1), 167–177 (1996)
10. Sullivan, D.: Measuring the degree of internationalization of a firm. *J. International Business Studies* 25(2), 325–342 (1994)
11. Taskaya-Temizel, T., Casey, M., Ahmad, K.: Pre-processing inputs for optimally configured time-delay neural networks. *IEE Electronics Letters* 41(4), 198–200 (2005)
12. Vicente, L., Vellido, A.: Review of Hierarchical Models for Data Clustering and Visualization, 12 pages (visited August 21, 2008), <http://www.lsi.us.es/redmidas/Capitulos/LMD20.pdf>
13. Yu, L., Wang, S., Lai, K.K.: Neural network-based mean-variance-skewness model for portfolio selection. *Computers and Operations Research* 35(1), 34–46 (2008)

An Adaptive Image Watermarking Scheme Using Non-separable Wavelets and Support Vector Regression

Liang Du¹, Xinge You¹, and Yiu-ming Cheung²

¹ Department of Electronics and Information Engineering
Huazhong University of Science and Technology, Wuhan, China

² Department of Computer Science,
Hong Kong Baptist University, Hong Kong SAR, China

aris-du@hotmail.com,
youxg@mail.hust.edu.cn,
ymc@comp.hkbu.edu.hk

Abstract. This paper presents an adaptive image watermarking scheme. Watermark bits are embedded adaptively into the non-separable wavelet domain based on the Human Visual System (HVS) model trained by Support Vector Regression (SVR). Unlike conventional separable wavelet filter banks that limit the ability in capturing directional information, non-separable wavelet filter banks contain the basis elements oriented at a variety of directions and different filter banks are able to capture different detail information. After removing the high frequency components, the low frequency subband used for watermark embedding is more robust against noise and other distortions. In addition, owing to the good generalization ability of the support vector machine, watermark embedding strength can be adjusted according to the HVS value. The superiority of non-separable wavelet transform (DNWT) in capturing image features combined with the good generalization ability of support vector regression provide us with a promising way to design a more robust watermarking algorithm featuring a better trade-off between the robustness and imperceptivity, the main duality of watermarking algorithms. Experimental results show that the DNWT watermarking scheme is robust to noising, JPEG compression, and cropping. In particular, it is more resistant to JPEG compression and noise than the discrete separable wavelet transform based scheme.

Keywords: Digital Non-tensor Product Wavelet Filters, Watermarking, Human Visual System, Support Vector Regression.

1 Introduction

With the rapid expansion of internet and wireless networks, multimedia security and digital rights management has been received much attention in the literature [1,2,5]. In general, a watermarking algorithm featuring robustness, perceptually invisibility, and security has been utilized to control the unauthorized duplication and redistribution of those multimedia data [2,3,4].

Traditionally, the wavelet based watermarking schemes employ the discrete separable wavelet transform (DSWT) to embed a watermark. However, the property of anisotropy makes the separable wavelet unattractive for the purpose of watermarking, which requires to extract more features of the image. In 1992, Jelena *Kovačević* [6] proposed to utilize the discrete non-separable wavelet transform (DNWT). Unlike the conventional separable wavelet filter banks that limit the ability in capturing directional information, non-separable wavelet filter banks contain the basis elements oriented at a variety of directions and different filter banks are able to capture the different detail information. In general, high frequency sub-bands of non-separable wavelet transform can reveal more features than that of the commonly used separable wavelet transform. After removing the high frequency components, the low frequency subband used for watermark embedding is more robust with regard to noise and other distortions. Nevertheless, it is a key issue how to construct the non-separable wavelet filter banks. Recently, You et.al have proposed a novel method to construct such filter banks in [7].

Furthermore, excessively higher embedding strength might lead to severe degradation of image quality. Human Visual System (HVS), which is always measured by Just Noticeable Difference (JND), should be taken into account while evaluating image quality [8,9]. Thus, it is necessary for watermarking algorithms to take the HVS into consideration. In addition, to solve the conflict between robustness and imperceptivity, some efforts have been made to utilize the machine learning technique for watermark embedding and extracting [10,11,12,13,14,15].

In this paper, we propose an adaptive image watermarking scheme based on discrete non-separable wavelet transform (DNWT) and support vector regression (SVR). The rest of this paper is organized as follows. The proposed algorithm is described in Section 2. We show our experimental results of the proposed scheme in Section 3. Finally, a conclusion is drawn in Section 4.

2 Watermarking Scheme Based on DNWT

Based on the constructed discrete non-separable wavelet filters in [7], we design the following algorithm. For a better tradeoff between the robustness and imperceptibility, the original image are decomposed into 3-levels in the proposed algorithm. Then, the watermarks are embedded into low frequency of the decomposed image. The overview of the proposed watermarking scheme is shown in Fig. (1). More details of the scheme will be described in the subsequent subsections.

2.1 HVS Modeling Using SVR

Selection of Features. A number of factors such as luminance, texture and edge affect the sensitivity of the human eyes to noise. In order to well model the

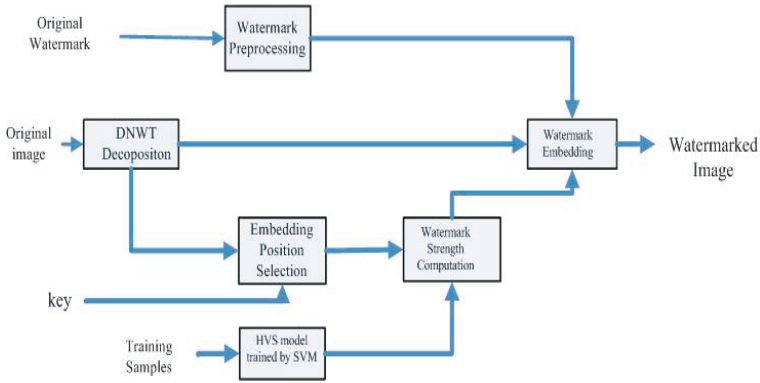


Fig. 1. The process of the watermark embedding based on DNWT and HVS

HVS, all these factors should be taken into account. In other words, features for training the SVR have to be selected appropriately.

Firstly, the impact of brightness on HVS is studied. Human eyes are less sensitive to the areas, where brightness is too high or too low. In this paper, the brightness sensitivity of an image with size $M \times N$ is measured as :

$$L = \sum_{x=1}^M \sum_{y=1}^N (f(x, y) - 128)^2 / \beta \tag{1}$$

where $f(x, y)$ denotes the grey level of the pixel in the x th row and the y th column, and β is a scaling factor. It can be seen that the higher or lower brightness pixel corresponding to lower L .

Secondly, let us take into account the local texture of an image. As illustrated in [16,17], human eyes are less sensitive to noise in highly textured areas, but not the edge areas. In general, entropy representing the turbidity of pixels can be utilized to measure human’s sensitivity to texture areas. In the smooth areas, the entropy value will be small. However, in both texture and edge areas, the entropy value becomes high although the sensitivity of human eyes to noise is quite different. Considering the measurement of edge areas, theories about edge detection might help. In this paper, the value of high frequency coefficients of wavelet domain is regarded as a way for representing edges. The number of high frequent coefficients whose absolute values greater than a threshold are defined as the measurement of edges. The entropy (ent) and edge(edg)can be calculated as follows:

$$ent = - \sum_{i=0}^{255} P(z_i) \log_2 P(z_i) \tag{2}$$

$$edg = \frac{U}{3 \sum_{n=1}^l M \times N / 2^n} \tag{3}$$

where z_i is the grey-level value of a pixel, $P(z_i)$ is a probability that the pixel value is equal to z_i , U denotes the number of coefficients whose absolute value is greater than the threshold T , and l is the decomposition level.

Acquisition of HVS Adaptive Watermark Strength by SVR. This paper utilizes the SVR to model the HVS in order to obtain an optimal adaptive watermark strength, i.e., the largest watermark strength can be achieved in the constraint of HVS(imperceptivity). The inputs of the SVR are luminance (L), the entropy (ent), and edge(edg), while the output is the optimal watermark strength. Specifically, the main learning steps are as follows:

- Let the training samples be:

$$T = \{(x_i, d_i) | i = 1, 2, \dots, l\}$$

where $x_i = (l_{i1}, ent_{i2}, edg_{i3}) \in R^3$ is an input vector, and d_i denotes the optimal watermark strength.

- In this paper, “RBF” Kernel of SVR is selected:

$$K(x, x_i) = (exp(-|x - x_i|^2/\sigma^2))$$

where σ is the width parameter of “RBF” kernel.

- The optimal model of SVR is as follows:

$$\text{Minimize } \frac{1}{2} \sum_{i,j=1}^l (\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j)K(x_i, x_j) + \varepsilon \sum_{i=1}^l (\alpha_i^* - \alpha_i) - \sum_{i=1}^l y_i(\alpha_i^* - \alpha_i)$$

$$\text{subject to } \sum_{i=1}^l (\alpha_i^* - \alpha_i) = 0, \frac{C}{T} \geq \alpha_i, \alpha_i^* \geq 0, i = 1, 2, \dots, l$$

where $\alpha_i, i = 1, 2 \dots l$ are the trained coefficients and C is the penalty parameter. Then, an optimal solution is

$$\alpha = (\alpha_1, \alpha_1^*, \dots, \alpha_l, \alpha_l^*)^T.$$

- The approximation of watermark strength function can be obtained:

$$d(i) = \sum_{i=1}^l (\alpha_i^* - \alpha_i)K(x_i, x) + b$$

where $b = y_i - \sum_{i=1}^l (\alpha_i^* - \alpha_i)(x_i \cdot x_j) + \varepsilon$ is the bias.

2.2 Watermark Embedding Method

Quantization Index Modulation (QIM), firstly proposed by Chen and Wornell [18], is applied to watermark adaptive embedding. Unlike traditional QIM whose quantization steps were identical, quantization steps are adaptively adjusted according to the HVS modeled by SVR. Different from the most adaptive algorithms

in which the embedding watermark bits are blockwise, the SVR input vectors are computed according to the quad-tree structure of multilevel decomposition of DNWT. Because it makes a good use of spatial localization of wavelet transform, the watermarking robustness improves. In some sense, the process of image watermarking is similar to image encoding. Both of them add a signal (e.g. watermark bits) or distortion (e.g. lossy compression) into an image. In the embedding phase of the proposed algorithm, we simulate the process of image coding with the purpose of reducing the probability that the watermark bits are lost as quantization noise. All these techniques are designed for prevent unintentional attacks like compression and noise pollution. Regarding intentional attacks, the key to selection of DNWT coefficients as well as the choice of parameters for non-sparable wavelet filters can act as private keys that prevent illegal users from extracting the watermarks even if the underlying watermarking algorithm is known. The proposed scheme is a blind watermarking algorithm, in which the watermarking procedure can be summarized as follows:

Step 1. Preprocessing of Watermarks

The two dimension binary watermark $W = \{w(i, j), i, j \in N, 1 \leq i \leq m, 1 \leq j \leq n\}$ is reshaped into one dimension vector $S = \{s(l), l \in N, 0 \leq l \leq m \times n\}$. We then encrypt and scramble it by a private key K_1 for security and the relativity reduction of cropping. If the mean of the watermark is smaller than 0.5, we reverse the binary image first.

Step 2. Selection of Embedding Coefficients

The original image is decomposed through DNWT filters into the three levels as shown in Fig. (2)(a). The coarse approximation subband denoted as A here is selected for watermark embedding. A pseudo-random number sequence P is generated by the key K_2 . We visit each coefficient in subband A columnwisely. If the i th element of P is 0, the corresponding coefficient is abandoned for watermark embedding.

Step 3. Computation of Adaptive Watermarking Strength

The quad-tree structure of multi-level DNWT decomposition which is the result of sub-sampling of DNWT decomposition is shown in Fig. (2)(b). The corresponding 8×8 block for each coefficient in spatial domain is used to compute the sensitivity of luminance L and entropy ent . The indicator of edge edg selects the high frequent wavelet coefficients in all three levels for computation. Three thresholds are needed for each subband. By using different level high frequent wavelet coefficients, the impacts of noise can therefore be reduced.

After all the three features are obtained, the embedding strength could be easily worked out according to the HVS model we have trained.

Step 4. Watermark embedding

For each selected coefficient $C_A(u_i, v_i)$, it is quantified as follows:

$$Q(C_A(u_i, v_i), d(i)) = \text{round}\left(\frac{C_A(u_i, v_i)}{d(i)}\right)d(i)$$

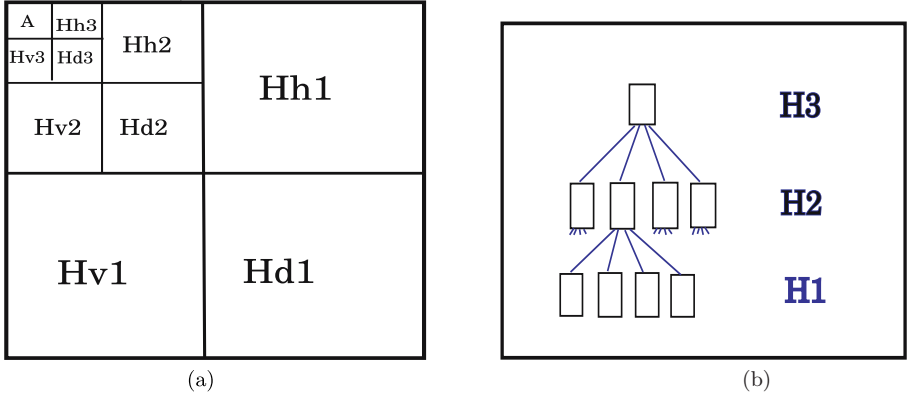


Fig. 2. Quad-tree structure of DNWT decomposition: (a) Sketch map of 3 level DNWT decomposition, (b) Quad-tree structure

where the function $round(\cdot)$ means rounding a value to the nearest integer. The watermark bit is embedded into the quantified wavelet coefficient as follows:

$$C_A(u_i, v_i)^* = Q(C_A(u_i, v_i) + s_i\beta, d(i)) - s_i\beta$$

with $\beta = d(i)/2$.

Step 5. Inverse DNWT

After all watermark bits are embedded, we apply the inverse DNWT to the modified coefficients. Then, the watermarked image is obtained.

2.3 Watermark Extraction Method

Extracting is the reverse phase of watermark embedding. The proposed scheme is a blind watermarking one, in which no original image is required during watermark extracting. The construction of wavelet filter and the private Key K are needed for extracting watermark bits. The extraction process is as follows:

Step 1. Decomposition of images

The watermarked image is decomposed by the 3-level DNWT using the same parameters A , B and N [7].

Step 2. Recovery of embedding positions

Find out the suspicious coefficients in subband A . After the same pseudo-random sequence is generated by the private key K_2 , positions in which watermark bits might have been embedded can be recovered.

Step 3. Watermarking Strength

For each coefficient, three features are calculated using (1), (2), (3) from the watermarked image. Using the HVS model trained by SVR, the corresponding watermarking strength $d(i)$ can be obtained.

Step 4. Watermark Bits Extraction

Let $\beta = d(i)/2$. Calculate two signals $S_r(0)$ and $S_r(1)$ by embedding “1” and “0” into the watermarked signal $C_A(u_i, v_i)^*$.

$$S_r(1) = Q(C_A(u_i, v_i)^* + \beta) - \beta, \quad S_r(0) = Q(C_A(u_i, v_i)^*).$$

The extracted message bit is then determined by judging which of these two signals has a smaller Euclidean distance, written as s_i , to the watermarked signal $C_A(u_i, v_i)^*$

$$s_i = \underset{m}{\operatorname{argmin}}(C_A(u_i, v_i)^* - S_r(m))^2, m \in 0, 1.$$

Then, the extracted bits is reshaped into the original watermark bits $W' = \{w(i, j), i, j \in N, 1 \leq i \leq m, 1 \leq j \leq n\}$ according to the private key K_1 .

To evaluate the performance of the proposed watermarking algorithm, we calculate the normalized correlation (NC) between the extracted watermark with the original one, i.e.

$$NC = \frac{\sum_{k=1}^{mn} W_k W'_k}{\sqrt{\sum_{k=1}^{mn} W_k^2} \sqrt{\sum_{k=1}^{mn} W'^2_k}}. \tag{4}$$

3 Experimental Results

In this section, we investigated on the robustness of our watermark scheme against JPEG compression, cropping, sharpening, and noise.

In our experiments, tested images were all standard images obtained from [19]. Fig. (4) presents the experimental results for our watermarking system based on DNWT against JPEG compression. The test image was a 512×512 LENA image. The size of watermark was 32×32 binary bits as shown in Fig. (3). Fig. (4) shows the NC value by the different A, B, N after JPEG compression attack. In the following, we let $A = 0.57$, $B = 0.78$, and $N = 3$. Cropping attacked images and watermarks extracted from them are shown in Fig. (5). It demonstrates that our proposed scheme is robust against cropping attacks. Further, Fig. (6) shows that our scheme is robust against the different noises such as “Gaussian” ”speckle”, and so on. Moreover, we investigated on the performance of our scheme against sharpening attacks. Fig. (7) shows the result.

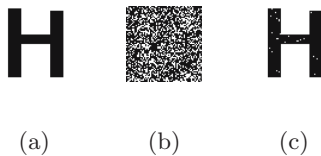


Fig. 3. Watermark image: (a) The original image; (b) The disturbed watermark; (c) The extracted watermark

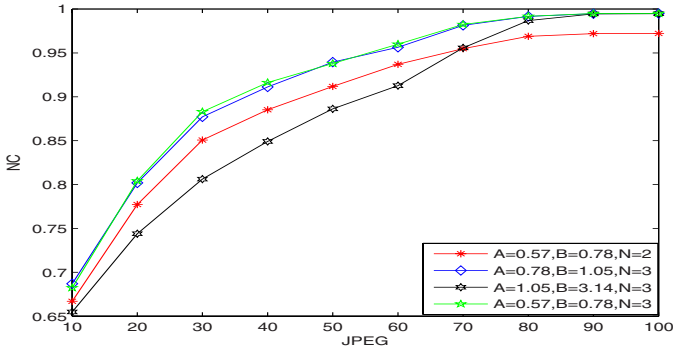


Fig. 4. NC after JPEG compression attack by the different values of A, B, N

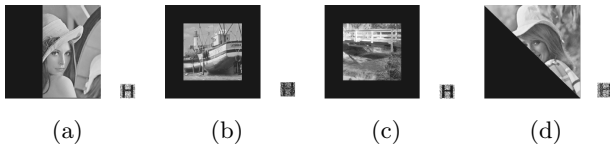


Fig. 5. Cropping attacked images and the watermark H extracted from them, respectively

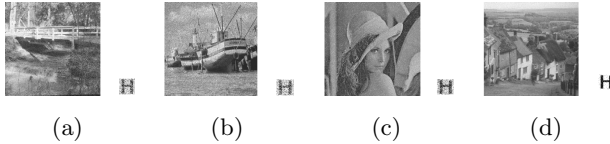


Fig. 6. Noising attacked images and the watermark H extracted from them, respectively, with (a) ‘Gaussian’ noise; (b) ‘speckle’ noise; (c) ‘Salt and pepper’ noise; (d) ‘Poisson’ noise

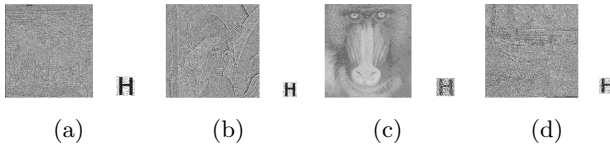


Fig. 7. Sharping attacked images and the watermark H extracted from them

4 Conclusion

This paper has proposed a DNWT based watermarking scheme, in which the watermark bits are embedded adaptively into the non-separable wavelet domain of an image based on the HVS model trained by the SVR. Our scheme features

the performance robustness against the JPEG compression, cropping, sharpening, and noise. The experiments have shown the promising results.

References

1. Lesk, M.: The good, the bad, and the ugly: What might change if we had good DRM. *IEEE Security & Privacy* 1(3), 63–66 (2003)
2. Hartung, F., Ramme, F.: Digital right management and watermarking of multimedia content for m-commerce applications. *IEEE Communications Magazine* 38(11), 78–84 (2000)
3. Yu, P.T., Tsai, H.H., Lin, J.S.: Digital watermarking based on neural networks for color images. *Signal Processing* 81, 663–671 (2001)
4. Wang, Y.W., Doherty, J.F., Van Dyck, R.E.: A wavelet-based watermarking algorithm for ownership verification of digital images. *IEEE Transactions on Image Processing* 11, 77–87 (2002)
5. Cox, I.J., Miller, M.L.: The first 50 years of electronic watermarking. *Journal on Applied Signal Processing* 2, 126–132 (2002)
6. Kovačević, J., Vetterli, M.: Non-separable multidimensional perfect reconstruction filter banks and wavelet bases for R_n . *IEEE Transactions on Information Theory* 38(2), 533–555 (1992)
7. You, X.G., Zhang, D., Chen, Q.H.: Face representation by using non-tensor product wavelets. In: *Proceedings of International Conference on Pattern Recognition*, pp. 503–506 (2006)
8. Watson, A.B.: DCT quantization matrices optimized for individual images. In: *Human Vision, Visual Processing, and Digital Display IV*, *Proceedings SPIE* 1913–1914, pp. 202–216 (1993)
9. Watson, A.B.: Visually optimal DCT quantization matrices for individual images. In: *Proceedings of IEEE Data Compression Conference*, pp. 178–187 (1993)
10. Lou, D.C., Liu, J.L., Hu, M.C.: Adaptive digital watermarking using neural network technique. In: *Proceedings of IEEE 37th Annual 2003 International Carnahan Conference on Security Technology*, vol. 37(10), pp. 325–332 (2003)
11. Yu, P.T., Tsai, H.H., Lin, J.S.: Digital watermarking based on neural networks for color images. *Signal Processing* 81, 663–671 (2001)
12. Zhang, J., Wang, N.C.: Neural network based watermarking for image authentication. *Journal of Computer-Aided Design & Computer Graphics* 15(3), 307–312 (2003)
13. Davis, K.J., Najarian, K.: Maximizing strength of digital watermarks using neural networks. In: *Proceedings of the IEEE International Joint Conference on Neural Networks*, pp. 2893–2898 (2001)
14. Li, C.H., Lu, Z.D., Zhou, K.: An image watermarking technique based on support vector regression. In: *Proceedings of ISCIT 2005*, pp. 177–180 (2005)
15. Li, J., Peng, H., Pei, Z.: Adaptive watermarking algorithm using SVR in wavelet domain. In: *6th IEEE/ACIS International Conference on Computer and Information Science*, pp. 207–211 (2007)
16. Huang, J.W., Yao, R.H.: Adaptive image watermarking algorithm. *Journal of Image and Graphics* 4, 640–643 (1999)

17. Yi, K.X., Shi, J.Y.: Adaptive 2-dimension image watermarking algorithm. *Journal of Image and Graphics* 6, 444–449 (2001)
18. Chen, B., Wornel, G.: An information-theoretic approach to design of robust digital watermarking systems. In: *Proceedings of IEEE International Conference on Acoustics, Speech and Singnal Processing*, vol. 4, pp. 2061–2064 (1999)
19. Petitcolas, F.A.P.: Weakness of Existing Watermark Scheme (1997), <http://www.petitcolas.net/fabien/watermarking/stirmark/index.html>

Cluster Analysis of Land-Cover Images Using Automatically Segmented SOMs with Textural Information

Márcio L. Gonçalves^{1,2}, Márcio L.A. Netto², and José A.F. Costa³

¹Department of Computer Science, PUC Minas, Poços de Caldas, MG, Brazil

²School of Electrical Engineering, State University of Campinas, Brazil

³Department of Electrical Engineering, Federal University of Rio Grande do Norte, Brazil
marcio@pucpcaldas.br, marcio@dca.fee.unicamp.br,
alfredo@dee.ufrn.br

Abstract. This work attempts to take advantage of the properties of Kohonen's Self-Organizing Map (SOM) to perform the cluster analysis of remotely sensed images. A clustering method which automatically finds the number of clusters as well as the partitioning of the image data is proposed. The data clustering is made using the SOM. Different partitions of the trained SOM are obtained from different segmentations of the U-matrix (a neuron-distance image) that are generated by means of mathematical morphology techniques. The different partitions of the trained SOM produce different partitions for the image data which are evaluated by cluster validity indexes. Seeking to guarantee even greater efficiency in the image categorization process, the proposed method extracts information from the image by means of pixel windows, in order to incorporate textural information. The experimental results show an application example of the proposed method on a TM-Landsat image.

Keywords: data clustering, self-organizing maps, image processing, remote sensing.

1 Introduction

The advances in computer and electronic technologies and decreasingly cost of memory storage systems have been enabling large amounts of data to be available in many application areas. An example is the remote sensing of the earth surface from satellite or airborne scanners. Large volumes of remotely sensed images are being generated from an increasing number of sophisticated airborne and space borne sensor systems, and while there is no substitute for a trained analyst, exploitation of this data on a large scale requires consistent automatic data exploration tools [8].

The self-organizing map (SOM), proposed by Kohonen [6], has been widely used in a variety of applications, including areas as data compression and data mining [7]. Important properties as the input space approximation, topological ordering and density matching, allied with the simplicity of the model and the easiness to implement its learning algorithm justify the success of the SOM and place it as one of the main models of neural nets in the present time.

This work presents a methodology that explores the characteristics and properties of the SOM to perform the cluster analysis of remotely sensed images. In the proposed method, the SOM is used to map the original patterns of the image to a 2-dimensional neural grid. The objective is to quantize and represent the image patterns in a space of smaller dimension, seeking to preserve the probability distribution and topology of the input space. Afterwards, different partitions of the trained SOM are obtained from different segmentations of the U-matrix [9], which are generated by means of mathematical morphology techniques. Each different partition of the U-matrix corresponds to a different clustering configuration of SOM neurons that can be utilized to represent the patterns by which the original image will be categorized. A cluster validity index is applied to determine automatically the best partition of the image data.

Seeking to guarantee even greater efficiency in the categorization process, the proposed method extracts samples from the image by means of pixel windows, in order to incorporate textural information. Following this approach, the method filters heterogeneous samples which represent patterns (pixel windows) corresponding to transition regions between different land cover classes.

The remainder of the paper is organized in the following form: section 2 describes succinctly the SOM; section 3 presents the proposed clustering methodology, while section 4 shows an application example of the proposed approach on a TM-LANDSAT image, and section 5 gives the conclusions and final considerations.

2 SOM

SOM is a type of artificial neural net based on competitive and unsupervised learning, i.e., no information about the input signal classes is used in the adjustment process for synaptic weight in the net [6]. The network essentially consists of two layers: an input layer I and an output layer U with neurons generally organized in a 2-dimensional topological array. The input to the net corresponds to a p -dimensional vector, \mathbf{x} , generally in the space \mathcal{R}^p . All of the p components of the input vector feed each of the neurons on the map. Each neuron i can be represented by a synaptic weight vector $\mathbf{w}_i = [w_{i1}, w_{i2}, \dots, w_{ip}]^T$, also in the p -dimensional space.

For each input pattern \mathbf{x} a winner neuron, c , is chosen, using the criterion of greatest similarity:

$$\|\mathbf{x} - \mathbf{w}_c\| = \min_i \{\|\mathbf{x} - \mathbf{w}_i\|\} \quad (1)$$

where $\|\cdot\|$ represents the Euclidian distance. The winner neuron weights, together with the weights of the neighboring neurons, are adjusted according to the following equation:

$$\mathbf{w}_i(t+1) = \mathbf{w}_i(t) + h_{ci}(t)[\mathbf{x}(t) - \mathbf{w}_i(t)] \quad (2)$$

where t indicates the iteration of the training process, $\mathbf{x}(t)$ is the input pattern and $h_{ci}(t)$ is the nucleus of neighborhood around the winner neuron c .

Once the SOM algorithm has converged, the 2-dimensional array of neurons displays important statistical properties, such as the approximation of the input space, topological ordering, and density matching.

Although the SOM presents attractive properties on the input data, the trained neural network requires additional procedures to enable a suitable interpretation of the data clusters. An example is the Unified Distance Matrix (U-matrix) method, which was developed by A. Ultsch [9] to detect non-linearities in the resulting SOM mapping. The basic idea is to use the same metric that was used during the learning to compute distances between adjacent reference vectors. This method can be used to visualize the topological structure of the SOM unit layer and therefore also the topology of the N -dimensional input space. The U-matrix can be visualized as a three dimensional landscape. Altitudes or the high places on the U-matrix encode dissimilarities between neurons and correspond to cluster borders while valleys represent to map units that are similar.

3 Proposed Clustering Methodology

The methodology proposed in this work essentially attempts to exploit the properties of SOM to perform the cluster analysis of remotely sensed images. The key strategy of the clustering method proposed here is to perform the clusters analysis of the image through a set of SOM prototypes instead of working directly with the original patterns of the scene. The method proposed basically presents four processing stages: sampling of the input image, training and segmentation of the SOM, and final categorization of the image.

In the following subsections each of the steps of the proposed methodology is explained in greater detail.

3.1 Sampling

The first step of the proposed methodology consists in collecting an image sample set in order to train the SOM. Unlike pixel by pixel approaches that only use the spectral information of individual points to find homogenous regions; the present work performs the sampling of the image through pixel windows. The idea is to incorporate in the clustering process information about the neighborhood (context) of the pixels, considering that isolated pixels are not able to represent the majority of cover land patterns, especially in the case of images that have higher spatial resolutions. The sample windows are collected uniformly across the entire region of the image, without overlappings and at regular intervals. All of the samples are square and have the same size.

Seeking to guarantee greater efficiency in the clustering method, the proposed method filters heterogeneous samples which represent patterns (pixel windows) corresponding to transition regions between different land cover classes. Heterogeneous samples are those that have a high degree of spectral heterogeneity and are normally associated with input patterns that incorporate more than one land cover class.

The spectral heterogeneity degree of each sample is computed from Haralick's co-occurrence matrix [4]. Since the samples are pixel windows, it makes it possible to generate an image of each sample and to calculate the co-occurrence probability of all pairwise combinations of grey levels in each one of them. The energy (sometimes called uniformity) was the measure chosen to calculate the spectral heterogeneity of

each sample from co-occurrence matrix. This measure, described through the equation (3), gets values next to 1 when the area of interest presents uniform texture (similar grey levels), and values that tend to zero when the area is not uniform.

$$ENE = \sum_i \sum_j (P(i, j)_{d, \theta})^2 \quad (3)$$

where $P(i, j)_{d, \theta}$ is the co-occurrence probability of two grey levels i and j , separate a distance d in the direction θ . The prototypes whose ENE's satisfy the relationship given below are considered heterogeneous and are consequently filtered:

$$ENE < \mu_{ENE} - \frac{1}{2} \sigma_{ENE} \quad (4)$$

Here μ_{ENE} and σ_{ENE} are, respectively the average and the standard deviation of the ENE's of all of the samples.

3.2 SOM Training

In order to train the SOM, some parameters must be specified to define the structure of the map and to specifically control the stated training. With the objective of guaranteeing good mapping of the original patterns, the proposed methodology defines in a particular way the neural net parameters based on the existing literature, on experimental tests, and some peculiarities of the application of SOM on remotely sensed images. However, since the SOM can be sensitive to choice of its training parameters, other alternatives can also be sought out to obtain good maps [5].

The proposed methodology utilizes the following parameters to train the SOM: linear initialization of weights, batch training mode, gaussian neighborhood function and rectangular shape to organize the two-dimensional array of neurons of the net.

The size of the map is one of the free parameters of SOM that particularly depends on the input image and the objectives of the clustering. If the objective is to detect all of the patterns in the image, including those with low probability of occurrence, large-sized maps must be employed in the analysis; in the opposite case, if the interest is concentrated only on the predominant patterns in the scene, a smaller-sized SOM can be utilized.

3.3 Segmentation of the SOM

At the third processing stage of the proposed approach the trained SOM is segmented. The strategy used in this work to interpret the SOM determines the best partition for the trained SOM from the analysis of different segmentations of the U-matrix. The strategy used can be seen as an improvement of the clustering method proposed in [2]. Costa and Netto [2] proposed an efficient method based on mathematical morphology to segment the U-matrix. The method applies the images segmentation algorithm, watershed [1], using a markers image to regularize the segmentation process. This same approach also is applied in our proposal, however, instead of using only one markers image to segment the U-matrix (as proposed in [2]), a quantity of markers images are considered and, therefore, different segmentations of the U-matrix are

obtained. Each one of these segmentations is associated with the neurons of the SOM, allowing to determine different partitions for the map which define different partitions for the image data. To select the best one among different partitions, each of these is evaluated using the CDbw cluster validity index proposed in [3].

Given the U-matrix image U obtained from the trained SOM, the following steps are performed to obtain the markers images set to the image U :

1. Filtering: create the image U_I by removing any pore with area less or equal than three pixels.
2. For $k = 1, \dots, f_{max}$, where f_{max} is the highest gray level of the image U_I {
 - 2.1. Create the binary image U_I^k that corresponds to the conversion U_I to a binary image using as threshold k .
 - 2.2. Obtain N_{rc}^k , the number of connected regions of U_I^k .}
3. Obtain the most persistent values of number of connected regions that correspond to the plateaus with sizes more than three contiguous gray levels in the plot of N_{rc}^k versus k .
4. Obtain the set of all markers images, $S_m = \{ U_2^{k_1}, U_2^{k_2}, \dots, U_2^{k_n} \}$, where k_1, k_2, \dots, k_n are initial values of the plateaus chosen in the previous step.

Although the used procedure to find the markers of the U-matrix is similar to that one presented in [2], in our approach the steps 4 and 5 determine a set of markers images instead of only one image considering all the significant plateaus.

Thus, the general strategy for the partitioning of the trained SOM can be summarized as follows:

1. Obtain the U-matrix using the trained SOM.
2. Find the set of all markers images to the U-matrix, $S_m = \{ U_2^{k_1}, U_2^{k_2}, \dots, U_2^{k_n} \}$ (as described previously).
3. For each markers image, $U_2^{k_i}, i=1,2,\dots,n$ {
 - 3.1. Compute the watershed lines on the U-matrix.
 - 3.2. Assign a different label for each connected region (cluster of neurons) of the U-matrix.
 - 3.3. Copy the U-matrix labels to the corresponding neurons in the map.
 - 3.4. Apply the CDbw index to evaluate the SOM partitioned in the step 3.4. }
4. Select the best partition for the SOM comparing the values of the CDbw index obtained for each different partition of the neurons map.

3.4 Image Categorization

In order to categorize the image, pixel windows are collected from the original image with equal dimensions from the training sample and are compared to all of the SOM prototypes. This comparison is performed through the distances calculated between the pixel windows and each of the prototypes. The central pixel of the pixel window receives the label of the prototype that has the least distance from it. The image is then entirely run through until all of the pixels have been labeled.

4 Experimental Results

This section shows an application example of the proposed methodology on a test image. The image used in the experiments is composed of spectral bands 3, 4, and 5 of the Landsat-5 TM satellite and was acquired on 20 August 1988. It has an IFOV of 30 m. The study area covers the city of Manaus and the intersection of two rivers, Rio Negro and Rio Solimões in the state of Amazonas, Brazil. Four large land cover patterns are present in the scene: urban area, vegetation, and two water patterns (the darker pattern corresponds to Rio Negro and the more purple pattern corresponds to Rio Solimões). This image was provided by the National Space Research Institute, (INPE), Brazil. The Fig.1 (a) shows a color composite of the test image.

Application of the proposed methodology was initially performed with a sampling process of the scene. Sample windows of size 5×5 were collected uniformly across the entire region of the image, without overlappings and at regular 10 pixel intervals, given a total of 2500 samples obtained without user intervention. Applying the samples filtering procedure, of the 2500 total samples, 635 of them presented a high degree of spectral heterogeneity, given that its ENE values exceeded the threshold defined in the equation (4), remaining then 1865 samples to be used in the SOM training stage.

A SOM composed of 144 neurons arranged in a 12×12 rectangular grid was trained with all of the 1865 samples. The other parameters of the SOM were defined according to the specifications presented in section 3.2.

The U-matrix was calculated from the trained SOM (Fig. 1(b) shows a 3D view of the U-matrix). The strategy proposed to segment the SOM (presented in the subsection 3.3) determines different partitions for the U-matrix using different markers images. It uses information like the number of connected regions (N_{rc}^k) for each gray level (k) of the U-matrix image for a useful gray level range, which in turn is related to the distances between neighboring neurons. For the U-matrix image of this experiment were determined four markers images. The markers images were obtained after thresholding the U-matrix by the lower gray levels from significant plateaus in the

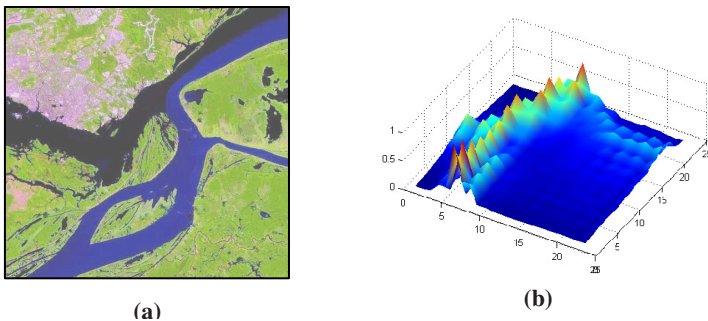


Fig. 1. (a) Color composite of the image used in the tests. (b) 3D view of the U-matrix for trained SOM.

plot of N_{rc}^k versus k (Fig. 2(a)). In this case, the values of these gray levels k were 14, 59, 101, and 138. Therefore, in accordance with the proposed method four different segmentations of U-matrix image were performed using the watershed method, one for each marker image. After, four different partitions for the SOM were determined from the segmented U-matrix images.

To evaluate the different partitions generated for the trained SOM, the CDbw cluster validity index was applied. The CDbw index presented the higher value for the partition $k = 49$, which corresponds to 4 clusters. For the CDbw index, the greater its value, the better the partition [3].

Figure 2(b) shows the test image categorized in accordance with the prototypes of the labeled SOM considering the partition $k = 49$.

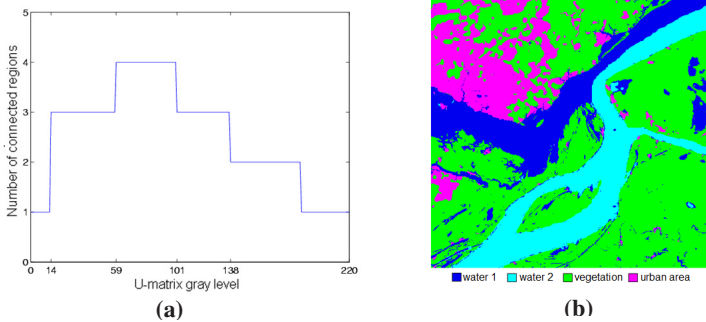


Fig. 2. (a) Number of connected regions versus image threshold of U-matrix. (b) Test image categorized by proposed method.

Attempting to evaluate the categorization generated by the proposed method, and considering the absence of terrestrial truth for the test image, the present work performed the classification of the image in a supervised manner using a multilayers Perceptrons (MLP) neural net, and considered these results as a reference (or “true”) to calculate the Kappa agreement index (normally used to evaluate the accuracy of satellite image categorization). The MLP net was trained with a sample set collected from original image by an image analyst. The Kappa index obtained here was 0.93, which allows concluding that the categorization result of the test image by method presented in this work was very satisfactory.

5 Conclusions and Final Considerations

In this work, a cluster analysis method of remotely sensed images that attempts to exploit the properties of the SOM was presented. The key point of the proposed method is to perform the clusters analysis of the image through a set of SOM prototypes instead of working directly with the original patterns of the scene. This approach significantly reduces the complexity of the analysis and presents advantages

that make it as a promising alternative to carry out the data clustering of remotely sensed images. Among these, we can point out:

- The proposed method does not require a previous definition of the number of clusters to perform the categorization of the image. It does not occur in the majority of the conventional unsupervised categorization methods;
- The distributed representation of the patterns by means of prototype groups gives the method the potential to discover geometrically complex and varied data clusters. Methods such as K-means use a single prototype (centroid) to represent each pattern and because of this are only capable of adequately detecting clusters that have hyperspherical formats;
- The simple use of pixel windows allows textural information to be included without any explicit calculation of measure for it. This approach contributes to the quality of the resulting categorization.

In addition to the test image utilized in the experiments shown here, the proposed method has also been applied to other high and medium resolution images, with satisfactory results.

References

1. Bleau, A., Leon, L.J.: Watershed-based segmentation and region merging. In: *Comp. Vis. Image Underst.*, vol. 77, pp. 317–370 (2000)
2. Costa, J.A.F., Netto, M.L.A.: Clustering of Complex Shaped Data Sets via Kohonen Maps and Mathematical Morphology. In: *Proceedings of the SPIE Conference on Data Mining and Knowledge Discovery*, Orlando, FL, vol. 4384, pp. 16–27 (2001)
3. Halkidi, M., Vazirgiannis, M.: Clustering validity assessment using multi representatives. In: *Proceedings of SETN Conference*, Thessaloniki, Greece (2002)
4. Haralick, R.M., Shanmugam, K., Dinstein, I.: Textural Features for Image Classification. *IEEE Trans. on Systems, Man and Cybernetics* 3(6), 610–621 (1973)
5. Kaski, S., Lagus, K.: Comparing self-organizing maps. In: Vorbrüggen, J.C., von Seelen, W., Sendhoff, B. (eds.) *ICANN 1996. LNCS*, vol. 1112, pp. 809–814. Springer, Heidelberg (1996)
6. Kohonen, T.: *Self-Organizing Maps*, 2nd edn. Springer, Berlin (1997)
7. Kohonen, T., Oja, E., Simula, O., Visa, A., Kangas, J.: Engineering Applications of the Self-Organizing Map. *Proceedings of the IEEE* 84(10), 1358–1384 (1996)
8. Richards, J.A.: Analysis of Remotely Sensed data: the formative decades and the future. *IEEE Transactions on Geoscience and Remote Sensing* 43, 422–432 (2005)
9. Ultsch, A.: Self-organizing neural networks for visualization and classification. In: *Information and Classification*, pp. 307–313. Springer, Berlin (1993)

Application of Topology Preserving Ensembles for Sensory Assessment in the Food Industry

Bruno Baruque¹, Emilio Corchado¹, Jordi Rovira², and Javier González²

¹ Department of Civil Engineering, University of Burgos, Spain
bbaruque@ubu.es, escorchado@ubu.es

² Department of Biotechnology and Food Science, University of Burgos, Spain
jrovira@ubu.es, javigonza77@hotmail.com

Abstract. Weighted Voting Superposition (WeVoS) is a novel summarization algorithm that may be applied to the results of an ensemble of topology preserving maps in order to identify the lowest topographical error in a map and thereby, to calculate the best possible visualization of the internal structure of its datasets. It is applied in this research to the food industry field that is studying the olfactory properties of Spanish dry-cured ham. The datasets used for the analysis are taken from the readings of an electronic nose, a device that can be used to recognize the sensory smellprints of Spanish dry-cured ham samples. They are then automatically analyzed using the previously mentioned techniques, in order to detect those batches with an anomalous smell (acidity, rancidity and different type of taints).. The Weighted Voting Superposition of ensembles of Self-Organising Maps (SOMs) is used here for visualization purposes, and is compared with the simple version of the SOM. The results clearly demonstrate how the WeVoS-SOM outperforms the simple SOM method.

1 Introduction

Topology preserving maps [1, 2] are often used for data visualization and inspection tasks. This interesting feature can assist human operators in classification tasks, such as the one presented in this study relating to the olfactory properties of Spanish dry-cured ham. Other features are pattern recognition and automated classification, inherent to many of the unsupervised learning techniques, which are especially relevant in the present application. These models are given enhanced stability in this study through the use of Weighted Voting Superposition (WeVoS), a novel ensemble summarization algorithm.

A combination of an electronic device for the analysis of volatile compounds (hereafter the electronic or “e-nose”) and a novel ensemble summarization algorithm for topology preserving mapping algorithms is used to study a wide variety of samples of “Serrano” Hams, in order to test whether this procedure is able to discriminate, in an easy and reliable way, between hams with different olfactory characteristics.

Consumer trust is a very important factor, when a product is being introduced into a new market or consolidated in an existing one. Dry-cured ham is a widely consumed traditional product in Spain that has also found a market outside Spain and is increasingly exported abroad. “Serrano Ham” is a salted ham that has been cured for over 210 days and is presented to the consumer on and off-the-bone. In these types of

products, rancid and acidic odours may be produced in storage; most of which may increase significantly because of proteolysis and lipid oxidation [3]. It is important to find quick and easy, low-cost techniques that apply simply parameters to evaluate the quality of these products prior to their sale and consumption by the consumer.

Several devices have appeared recently with the aim of enabling analytical techniques in the food industry to support the subjective decisions of professional testers. One disadvantage of these alternative tests is that whatever humans interpret as tastes and smells, machines will interpret as inevitably complex, numeric measurements. Thus, the aim of this multidisciplinary research is to devise an artificial intelligence system capable of interpreting the analyses made by an e-nose and presenting the results in an easily understandable way to human experts.

The rest of this paper is organized as follows: Section 2 and Section 3 present the AI models used in this research. Section 4 outlines the data gathering and pre-processing of the information in the samples, while Section 5 describes the experiments and results and finally, Section 6 presents the conclusions and future lines of research.

2 Topology Preserving Models

Topology preserving mapping comprises a family of techniques with a common target: to produce a low-dimensional representation of the training samples that preserves the topological properties of the input space. From among the various techniques, the best known is the Self-Organizing Map (SOM) algorithm [1, 2]. SOM aims to provide a low-dimensional representation of multi-dimensional datasets while preserving the topological properties of the input space. The SOM algorithm is based on competitive unsupervised learning; an adaptive process in which the neurons in a neural network gradually become sensitive to different input categories, which are sets of samples in a specific domain of the input space [4].

The update of neighbourhood neurons in SOM is expressed as:

$$w_k(t+1) = w_k(t) + \alpha(t)\eta(v,k,t)(x(t) - w_v(t)) \quad (1)$$

where w_v is the winning neuron, α is the learning rate of the algorithm, and $\eta(v,k,t)$ the neighbourhood function, in which v represents the position of the winning neuron in the lattice and k the positions of the neurons in the neighbourhood of this one, and x , the network input.

This model can be adapted for classification of new samples using a semi-supervised procedure [5].

3 Ensembles of Topology Preserving Maps

The idea behind the novel fusion algorithm, WeVoS, is to obtain a final map keeping one of the most important features of these types of algorithms: its topological ordering. WeVoS is an improved version of an algorithm presented in several previous works: [6, 7] and in this study is applied for the first time to the SOM in the field of the food industry.

It is based on the calculation of the “quality of adaptation” of a homologous unit of different maps, in order to obtain the best characteristics of the vector in each of the units that make up the final map. This calculation is performed as follows:

$$V(p) = \frac{|x_p|}{\sum |x_p|} \cdot \frac{q_p}{\sum q_p} \quad (2)$$

In this study, two slightly different versions of the WeVoS meta-algorithm are compared: WeVoS (pos) and WeVoS (map). They differ in the way that they consider map units as “homologous” to the summary map units that they calculate. The first version, WeVoS (pos), considers the units that have been assigned to the same position in different maps as homologous. The second, WeVoS (map), considers other units in the neighbourhood of that unit in the same map as homologous.

The general WeVoS meta-algorithm is described in detail in *Algorithm 1*.

Algorithm 1. Weighted Voting Superposition (WeVoS)

```

1: train several networks by using the bagging (re-sampling with replacement) meta-
algorithm
2: for each map (m) in the ensemble
3: for each unit position (p) of the map
4: calculate the quality measure/error chosen for the current unit
5: end
6: end
7: calculate an accumulated quality/error total for each homologous set of units  $Q(p)$  in all
maps
8: calculate an accumulated total of the number of data entries recognized by an homolo-
gous set of units in all maps  $D(p)$ 
9: for each unit position (p)
10: initialize the fused map (fus) by calculating the centroid ( $w'$ ) of the units of all maps in
that position (p)
11: end
12: for each map (m) in the ensemble
13: for each unit position (p) of the map
14: calculate the vote weight of the neuron (p) in the map (m) by using Eq. 2
15: feed the weight vector of the neuron (p), as if it were a network input, into the fused map
(fus), using the weight of the vote calculated in Eq 2 as the learning rate and the index of
that same neuron (p) as the index of the BMU.
The unit of the composing ensemble ( $w_p$ ) is thereby approximated to the unit of the final
map ( $w'$ ) according to its weighting system.
16: end
17: end

```

4 A Food Industry Case Study

4.1 Preliminary Analysis of the Ham Samples

Several Spanish hams of different qualities and origins were used in this research. The data sets consisted of measurements taken from seven types of Spanish dry-cured ham from among the various brands available on the Spanish market. The samples also

included some that were tainted and/or that had a rancid/acidic taste. The tainted samples were randomly taken from among all the different quality types and origins of hams. The commercial brands of the hams in the samples were not taken into account in this study.

In this case the e-nose was used to measure the odour of the ham samples. The data collected was presented to the ensemble summarization algorithm of topology preserving maps, WeVoS, in order to achieve a simple and reliable device for testing and analysing the olfactory properties of the hams.

4.2 E-Nose Odour Recognition

The odour recognition process may be summarized as follows:

1. The sample is heated for a given time to generate volatile compounds in the head-space of the vial containing the sample.
2. The gas phase is transferred to a detection device which reacts to the presence of molecules.
3. The differences in sensor reactions are recorded using statistical calculation techniques to classify the odours. The readings taken by each sensor are separated and stored in a simple database for further study.

In this study the analyses are performed using an E-Nose α FOX 4000 (Alpha M.O.S., Toulouse, France) with a sensor array of 18 metal oxide sensors. The e-nose takes readings every 0.5 seconds, and has an acquisition time of 120 seconds and an acquisition delay of 600 seconds. Only the highest reading from each sensor is stored in the database for further analysis.

5 Empirical Evaluation

After having obtained the readings for each sample of cut ham taken from the 18-sensor array in the electronic nose, they are stored in a database along with the corresponding results of the sensory evaluation by the professional testers. These results are normally more detailed, but were restricted in this initial study to three possible values: “unspoilt”, “rancid/acid” and “tainted”. Thus, our final dataset consisted of readings taken from a total of 154 samples of ham, the readings on each ham being composed of 18 different variables measured over three possible categories.

Regarding the visual inspection of the maps obtained in Figs. 1(a) to 1(d), the projection of the dataset over its two first Principal Components (obtained by a conventional PCA analysis [8]) is shown alongside three maps obtained by training over the same dataset. If attention is paid to Fig. 1(a), it may be observed that the dataset is clearly ordered. Most of the unspoilt samples are situated in a compact group on the right of the image (triangles), while tainted samples are represented in 2 main groups to the left of the image (circles). The rancid/acidic samples (squares), although they might have been considered unspoilt, were on the point of spoiling. In Fig. 1(a) they are clearly shown as separate from the group of the normal samples and lie within the group of definitively tainted samples.

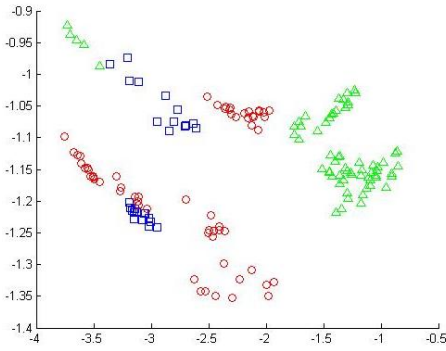


Fig. 1(a). Ham dataset projection over the first 2 Principal Components

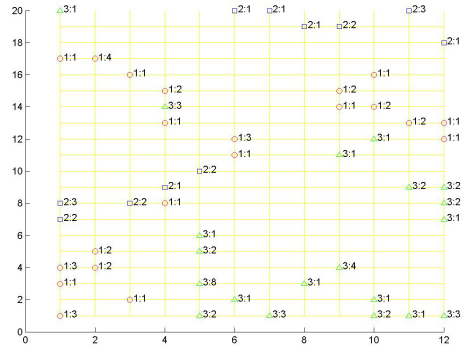


Fig. 1(b). Map obtained by training a single SOM over the ham dataset

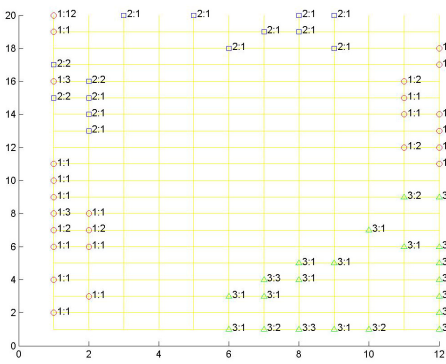


Fig. 1(c). Map obtained by calculating a WeVoS-SOM (pos) from an ensemble of SOMs trained over the ham dataset

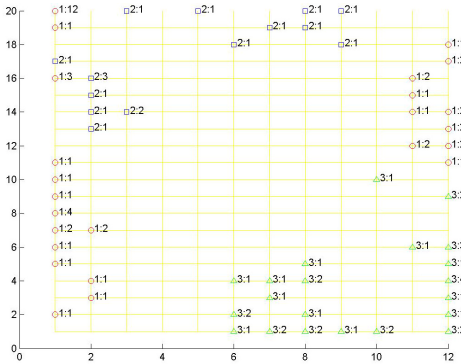


Fig. 1(d). Map obtained by calculating a WeVoS-SOM (map) from an ensemble of SOMs trained over the ham dataset

This same organization can be observed in the maps representing the dataset, although it is much less clearly represented in Fig. 1(b), which depicts the map obtained by a single SOM. The samples are scattered across this map, and there are even some unspoil samples among the spoil ones. On the contrary, in Figs. 1(c) and 1(d), the data appears more ordered, with all the samples in the unspoil group to the bottom right-hand corner, clearly separated from the rest. Those maps even represent the gap separating the two groups of tainted samples, which emerges due to the different origin of the samples: one group is composed of samples originally of high quality hams that became spoiled, while the other is composed of samples of standard quality hams that also became spoiled. This situation is less evident if the only map observed is the one obtained from a single model (Fig. 1(b)).

The next step in the study was the training of single maps and ensembles of a different number of maps over the same subset of samples, in order to compare their characteristics. This was done using a standard 5-fold cross-validation technique in

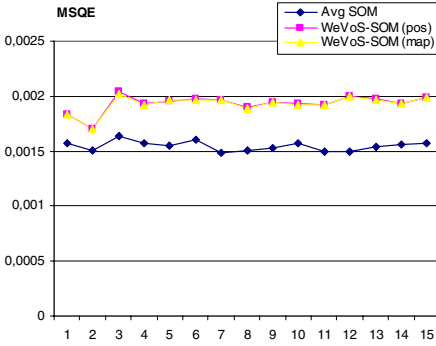


Fig. 2 (a). Mean Square Quantization Error

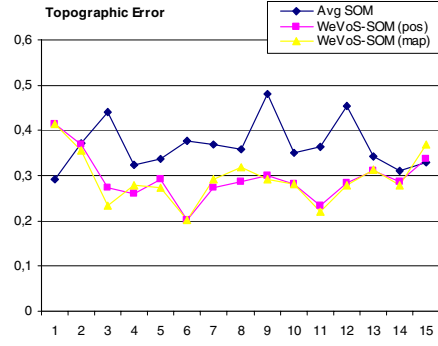


Fig. 2 (b). Topographic Error

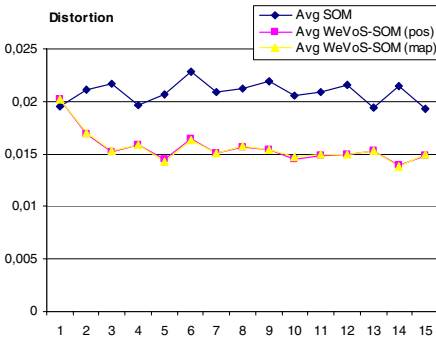


Fig. 2 (c). Distortion

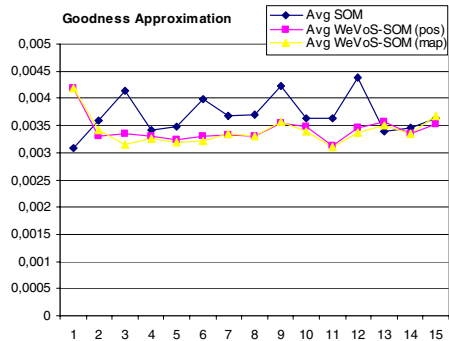


Fig. 2 (d). Goodness of Approximation

Fig. 2. Error readings for each of the models compared in the study (single map, WeVoS (pos) and WeVoS (map)). They were all obtained from the same basic SOM model. The x-axis represents the number of maps used in the ensemble and the y-axis the value of the measure.

order to be able to use the whole dataset for the tests. Each measure obtained represented the average of the measures obtained by each of the maps trained with 4-folds and tested over the other remaining fold.

Fig. 2 shows several measurements obtained from three models compared in the study, all of which are error measurements in different areas of representation of the dataset. The definitions of the Mean Square Quantization Error (MSQE), Topographic Error and Distortion are found in [9], while the Goodness of Approximation is described in [10]. As may be observed in the last three measurements (Fig. 2(b) to Fig. 2(d)), not only do the ensemble summarization methods (WeVoS) obtain a lower error, but they are also more stable, and do not depend on any specific execution of the training. As expected, the only exception to this is the MSQE, because it is a measurement of how far or how close the samples are from the unit that represents them, whereas the WeVoS meta-algorithm improves the visual representation of the dataset; the remaining measurements denote the accuracy of the representation in relation to the topological organization of the map.

6 Conclusions and Future Work

It has been shown that the summarization algorithm presented in this article is capable of providing a better visualization than the simple version of the SOM model.

In this case, it has been successfully applied to the readings taken from an E-nose in an assessment of the olfactory properties of different ham samples. The results suggest that this combination of techniques may easily be adapted to assist professional food testers in their work of classifying food samples or may even replace them in cases where simple explanations of taste are required.

Future work will include a more thorough study and classification of the samples in order to provide the professional food tester with maps that include more detailed information on the quality of the samples provided. Another line of work consists in adapting the algorithm to other extensions of the SOM to improve visualization, such as the Visualization Induced SOM (ViSOM).

Acknowledgements

This research has been partially funded through project BU006A08 of the JCyL.

References

- [1] Kohonen, T.: *Self-Organizing Maps*. Springer, Berlin (1995)
- [2] Kohonen, T.: The self-organizing map. *Neurocomputing* 21, 1–6 (1998)
- [3] Monahan, R.L., Brunton, N.P., Cronin, D.A., Durcan, R.: Determination of hexanal in cooked turkey using solid phase microextraction (SPME)/GC). In: 44th International Congress of Meat Science and Technology (ICoMST), vol. 1, pp. 586–587 (1998)
- [4] Kohonen, T., Lehtio, P., Rovamo, J., Hyvarinen, J., Bry, K., Vainio, L.: A principle of neural associative memory. *Neuroscience* 2, 1065–1076 (1977)
- [5] Kraaijveld, M.A., Mao, J., Jain, A.K.: A nonlinear projection method based on Kohonen's topology preserving maps. *IEEE Transactions, Neural Networks* 6, 548–559 (1995)
- [6] Corchado, E., Baruque, B., Yin, H.: Boosting Unsupervised Competitive Learning Ensembles. In: *International Conference of Neural Network (ICANN 2007)*, pp. 339–348 (2007)
- [7] Baruque, B., Corchado, E., Yin, H.: Quality of Adaptation of Fusion ViSOM. In: Yin, H., Tino, P., Corchado, E., Byrne, W., Yao, X. (eds.) *IDEAL 2007*. LNCS, vol. 4881, pp. 728–738. Springer, Heidelberg (2007)
- [8] Hotelling, H.: Analysis of a Complex of Statistical Variables into Principal Components. *Journal of Education Psychology* 24, 417–444 (1933)
- [9] Pözlbauer, G.: Survey and Comparison of Quality Measures for Self-Organizing Maps. In: *Fifth Workshop on Data Analysis (WDA 2004)*, pp. 67–82 (2004)
- [10] Kaski, S., Lagus, K.: Comparing Self-Organizing Maps. In: Vorbrüggen, J.C., von Seelen, W., Sendhoff, B. (eds.) *ICANN 1996*. LNCS, vol. 1112, pp. 809–814. Springer, Heidelberg (1996)

AI for Modelling the Laser Milling of Copper Components

Andrés Bustillo¹, Javier Sedano², José Ramón Villar³, Leticia Curiel¹,
and Emilio Corchado¹

¹ Department of Civil Engineering, University of Burgos, Burgos, Spain

² Department of Electromechanical Engineering, University of Burgos, Burgos, Spain

³ Department of Computer Science, University of Oviedo, Spain

abustillo@ubu.es, jsedano@ubu.es, escorchado@ubu.es,
lcuriel@ubu.es, villarjose@uniovi.es

Abstract. Laser milling is a relatively new micromanufacturing technique in the production of copper and other metallic components. This study presents multidisciplinary research, which is based on unsupervised connectionist architectures in conjunction with modelling systems, on the determination of the optimal operating conditions in this industrial process. Sensors on a laser milling centre relay the data used in this industrial case study of a machine-tool that manufactures copper components for high value micro-coolers. The two-phase application of the connectionist architectures is capable of identifying a model for the laser-milling process based on low-order models such as Black Box. The final system is capable of approximating the optimal form of the model. Finally, it is shown that the Box-Jenkins algorithm, which calculates the function of a linear system from its input and output samples, is the most appropriate model to control these industrial tasks.

1 Introduction

Laser milling of copper is a complicated process due to the high conductivity and high reflectivity of this metal. Laser milling, in general, consists of the controlled evaporation of waste material due to its interaction with high-energy pulsed laser beams. The operator of a conventional milling machine is aware at all times of the amount of waste material removed, but the same can not be said of a laser milling machine. A model that could predict the exact amount of material that each laser pulse is able to remove would contribute to the industrial use and development of this new technology. The one proposed in this paper is able to optimize the manufacturing process and to control laser milling to a level of accuracy that is required for the manufacture of micro-coolers. It has been developed using a combination of conventional and Artificial Intelligence (AI) models and is applied here to data taken from micromanufacturing laser milling of copper components.

Unsupervised neural networks can be used as a preliminary phase or step before a model is established. They are used to analyze the internal structure of the data sets in order to establish that they are sufficiently informative.

The rest of the paper is organized as follows. Following the introduction, a two-phase process is described to identify the optimal conditions for the industrial laser

milling of copper components. The case study is then presented that outlines the practical application of the model. Finally, some of the different modelling systems are applied and compared, in order to select the best model in this case, before ending with a short conclusion that summarises the salient points of this work.

2 Modelling the Laser Milling of Copper Components

2.1 A First Phase Using Connectionist Models

Cooperative Maximum-Likelihood Hebbian Learning (CMLHL) [2] is applied in this study in order to analyse the internal structure of the data set under study and to establish whether it is “sufficiently informative”. In the worse case, the experiments have to be performed again.

CMLHL is a Exploratory Projection Pursuit (EPP) method [1] [3], [4]. In general, EPP provides a linear projection of a data set, but it projects the data onto a set of basic vectors which help reveal the most interesting data structures; interestingness is usually defined in terms of how far removed the distribution is from the Gaussian distribution [5].

One connectionist implementation is Maximum-Likelihood Hebbian Learning (MLHL) [4], [6]. It identifies interestingness by maximising the probability of the residuals under specific probability density functions that are non-Gaussian. An extended version is the CMLHL [2] model, which is based on MLHL [4],[6] but adds lateral connections [7], [2] that have been derived from the Rectified Gaussian Distribution [5].

Considering an N-dimensional input vector (x), and an M-dimensional output vector (y), with W_{ij} being the weight (linking input j to output i), then CMLHL can be expressed [8], [9] as:

1. Feed-forward step:

$$y_i = \sum_{j=1}^N W_{ij} x_j, \forall i \quad (1)$$

2. Lateral activation passing:

$$y_i(t+1) = [y_i(t) + \tau(b - Ay)]^+ \quad (2)$$

3. Feedback step:

$$e_j = x_j - \sum_{i=1}^M W_{ij} y_i, \forall j \quad (3)$$

4. Weight change:

$$\Delta W_{ij} = \eta \cdot y_i \cdot \text{sign}(e_j) |e_j|^{p-1} \quad (4)$$

Where: η is the learning rate, τ is the "strength" of the lateral connections, b the bias parameter, p a parameter related to the energy function [2], [4], [6] and A a symmetric matrix used to modify the response to the data [2]. The effect of this matrix is based on the relation between the distances separating the output neurons.

2.2 Second Phase

The **identification criterion** evaluates which of the group of candidate models is best adapted to and which best describes the data sets collected in the experiment; i.e., given a model $M(\theta_*)$ its prediction error may be defined by equation (5); and a good model [8] will be that which makes the best predictions, and which produces the smallest errors when compared against the observed data. In other words, for any given data group Z' , the ideal model will calculate the prediction error $\mathcal{E}(t, \theta)$, equation (5), in such a way that for any one $t=N$, a particular $\hat{\theta}_N$ (estimated parametrical vector) is selected so that the prediction error $\mathcal{E}(t, \hat{\theta}_N)$ in $t=1,2,3\dots N$, is made as small as possible.

$$\mathcal{E}(t, \theta_*) = y(t) - \hat{y}(t | \theta_*). \tag{5}$$

The estimated parametrical vector $\hat{\theta}$ that minimizes the error, equation (8), is obtained from the minimization of the error function (6). This is obtained by applying the least-squares criterion for the linear regression, i.e., by applying the quadratic norm $\ell(\mathcal{E}) = \frac{1}{2} \mathcal{E}^2$, equation (7).

$$V_N(\theta, Z^N) = \frac{1}{N} \sum_{t=1}^N \ell(\mathcal{E}_F(t, \theta)). \tag{6}$$

$$V_N(\theta, Z^N) = \frac{1}{N} \sum_{t=1}^N \frac{1}{2} (y(t) - \hat{y}(t | \theta))^2. \tag{7}$$

$$\hat{\theta} = \hat{\theta}_N(Z^N) = \underset{\theta \in D_M}{\text{arg min}} V_N(\theta, Z^N). \tag{8}$$

The methodology of black-box structures has the advantage of only requiring very few explicit assumptions regarding the pattern to be identified, but that in turn makes it difficult to quantify the model that is obtained. The discrete linear models may be represented through the union between a deterministic and a stochastic part, equation (9); the term $e(t)$ (white noise signal) includes the modelling errors and is associated with a series of random variables, of mean null value and variance λ .

$$y(t) = G(q^{-1})u(t) + H(q^{-1})e(t). \tag{9}$$

The structure of a black-box model depends on the way in which the noise is modelled $H(q^{-1})$; thus, if this value is 1, then the OE (Output Error) model is applicable; whereas, if it is different from zero a great range of models may be applicable; one of the most common being the BJ (Box Jenkins) algorithm. This structure may be represented in the form of a general model, where $B(q^{-1})$ is a polynomial of grade nb , which can incorporate pure delay nk in the inputs, and $A(q^{-1}), C(q^{-1}), D(q^{-1})$ and $F(q^{-1})$ are autoregressive polynomials ordered as na, nc, nd, nf , respectively (10).

Likewise, it is possible to use a predictor expression, for the on-step prediction ahead of the output $\hat{y}(t|\theta)$ (11). In Table 1, the generalized polynomial expressions are presented, as well as those that represent the polynomials used in the case of each particular model.

$A(q^{-1})y(t) = q^{-n_k} \frac{B(q^{-1})}{F(q^{-1})}u(t) + \frac{C(q^{-1})}{D(q^{-1})}e(t)$	(10)
$\hat{y}(t \theta) = \frac{D(q^{-1})B(q^{-1})}{C(q^{-1})F(q^{-1})}u(t) + \left[1 - \frac{D(q^{-1})A(q^{-1})}{C(q^{-1})}\right]y(t)$	(11)

Table 1. Black-box model structures

Polynomials in (10)	Polynomials used in (10)	Name of model structure
$A(q^{-1}) = 1 + a_1(q^{-1}) + a_2(q^{-2}) + \dots + a_{n_a}(q^{-n_a})$	B AB ABC AC BF BFCD	FIR
$B(q^{-1}) = b_1(q^{-1}) + b_2(q^{-2}) + \dots + b_{n_b}(q^{-n_b})$		ARX
$C(q^{-1}) = 1 + c_1(q^{-1}) + c_2(q^{-2}) + \dots + c_{n_c}(q^{-n_c})$		ARMAX
$D(q^{-1}) = 1 + d_1(q^{-1}) + d_2(q^{-2}) + \dots + d_{n_d}(q^{-n_d})$		ARMA
$F(q^{-1}) = 1 + f_1(q^{-1}) + f_2(q^{-2}) + \dots + f_{n_f}(q^{-n_f})$		OE BJ

Procedure for Modelling the Laser Milling process. The identification procedure used to arrive at a parameterized model M, which will eventually be selected as the best from among those that modelled the laser milling characteristics on the basis of the variable measurements, is carried out in accordance with two fundamental patterns: a first pre-analytical and then an analytical stage that assists with the determination of the parameters in the identification process and the model estimation. The pre-analysis test is run to establish the identification techniques [8], [9], [10], [11], [12], [13], the selection of the model structure and its order estimation [14], [15], the identification criterion and search methods that minimize it and the specific parametrical selection for each type of model structure.

A second validation stage ensures that the selected model meets the necessary conditions for estimation and prediction. Three tests were performed to validate the model: residual analysis $\mathcal{E}(t, \hat{\theta}(t))$, by means of a correlation test between inputs, residuals and their combinations; final prediction error (FPE) estimate, as explained by Akaike [16]; and the graphical comparison between desired outputs and the outcome of the models through simulation one (or k) steps before.

3 A Case Study: Laser Milling of Copper Components

This multidisciplinary work sets out to study and identify the optimal conditions for laser milling of computer components in a micromanufacturing technique to produce

micro-coolers that uses a commercial Nd:YAG laser with a pulse length of $10\mu\text{s}$. Three parameters of the laser process can be controlled: laser power (u_1), laser milling speed (u_2) and laser pulse frequency (u_3). The laser is integrated in a laser milling centre (DMG Lasertec 40).

To simplify this industrial problem a test piece was designed and used in all of the laser milling experiments. It consisted on an inverted, truncated, pyramid profile that had to be laser milled on a flat metallic piece of copper. The truncated pyramid had angles of 135° , and a depth of 1 mm, but as the optimized parameters for the laser milling of the copper were not known at that point in time, both parameters showed errors, which are referred to, in this paper, as angle error (y_1) and depth error (y_2). A third parameter to be considered was the surface roughness of the milled piece (y_3). This variable also had to be optimized, because the industrial process required a precise geometrical shape, but also a good surface roughness of the piece. We applied different modelling systems to achieve the optimal conditions of these three parameters.

Table 2. Variables, units and values used during the experiments. All values are common to this laser milling process. Output $y(t)$, Input $u(t)$.

Variable (Units)	Range
o Angle error of the test piece, $y_1(t)$	-1 to 1
o Depth error of the test piece, $y_2(t)$	-1 to 1
o Surface roughness of the test piece (μm), $y_3(t)$	0.8 to 15
o Laser power in percent of the maximum power performed by the laser (%), $u_1(t)$.	20 to 100
o Laser milling speed (mm/s), $u_2(t)$.	200 to 800
o Laser pulse frequency (kHz), $u_3(t)$.	20 to 100

The experimental design was performed on a Taguchi L25 with 3 input parameters and 5 levels, so as to include the entire range of laser milling settings that are controllable by the operator. Table 2 summarizes the input and output variables of the experiment. The experiment was performed on the test piece described above. After the laser milling, actual inverted pyramid depth, wall angle and surface roughness of the bottom surface were measured using optical devices. The two first measurements were compared with the nominal values in the CAD model, thereby obtaining the two errors (y_1 and y_2). The test piece and the prototype were described in detail beforehand [17].

3.1 Application of the Two Phases of the Modelling System

The experiments have been organized into two phases.

- Phase 1. Initial identification of the internal structure of the data set. Application of several unsupervised neural models.
- Phase 2. Final identification of the model that best defines the dynamic of the laser milling process.

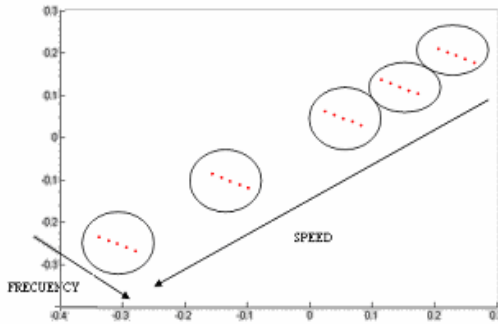


Fig. 1. The first of two projections obtained by CMLHL

Phase 1. Figure 1 shows the results obtained from the first two CMLHL projections. We can see how this method identifies a clear structure of five clusters ordered by speed and by frequency, which indicates that the data analysed is sufficiently informative.

Phase 2. Modelling the laser milling process. Figure 2, shows the results of output $y_1(t)$, angle error, $y_2(t)$, depth error and $y_3(t)$ surface roughness, respectively, for the different models. They show the graphic representations of the results, for ARX models, in relation to the polynomial order and the delay in the inputs; various delays for all inputs and various polynomial orders [na, nb₁, nb₂, nb₃, nk₁, nk₂, nk₃] were considered to arrive at the highest degree of precision, in accordance with the structure of the models that have been used; see Table 1. In Fig. 2, the X-axis shows the number of samples used in the validation of the model, while the Y-axis represents the range of output variables.

Table 3 shows a comparison of the qualities of estimation and prediction of the models obtained, as a function of the model, the estimation method, and the indicators, which are defined as follows:

- The percentage representation of the estimated model (expressed as so many percent “%”) in relation to the true system: the numeric value of the normalized mean error that is computed with one-step prediction (FIT1), with ten-step prediction (FIT10), or by means of simulation (FIT). Also shown are the graphical representations of true system output and both the one-step prediction $\hat{y}_1(t|m)$, the ten-step prediction $\hat{y}_{10}(t|m)$, and the model simulation $\hat{y}_\infty(t|m)$.
- The loss or the error function (V): the numeric value of the mean square error that is calculated from the estimation data set.
- The generalization error value (NSSE): the numeric value of the mean square error that is calculated from the validation data set.
- The average generalization error value (FPE): This is the numeric value of the FPE criterion that is calculated from the estimation data set.

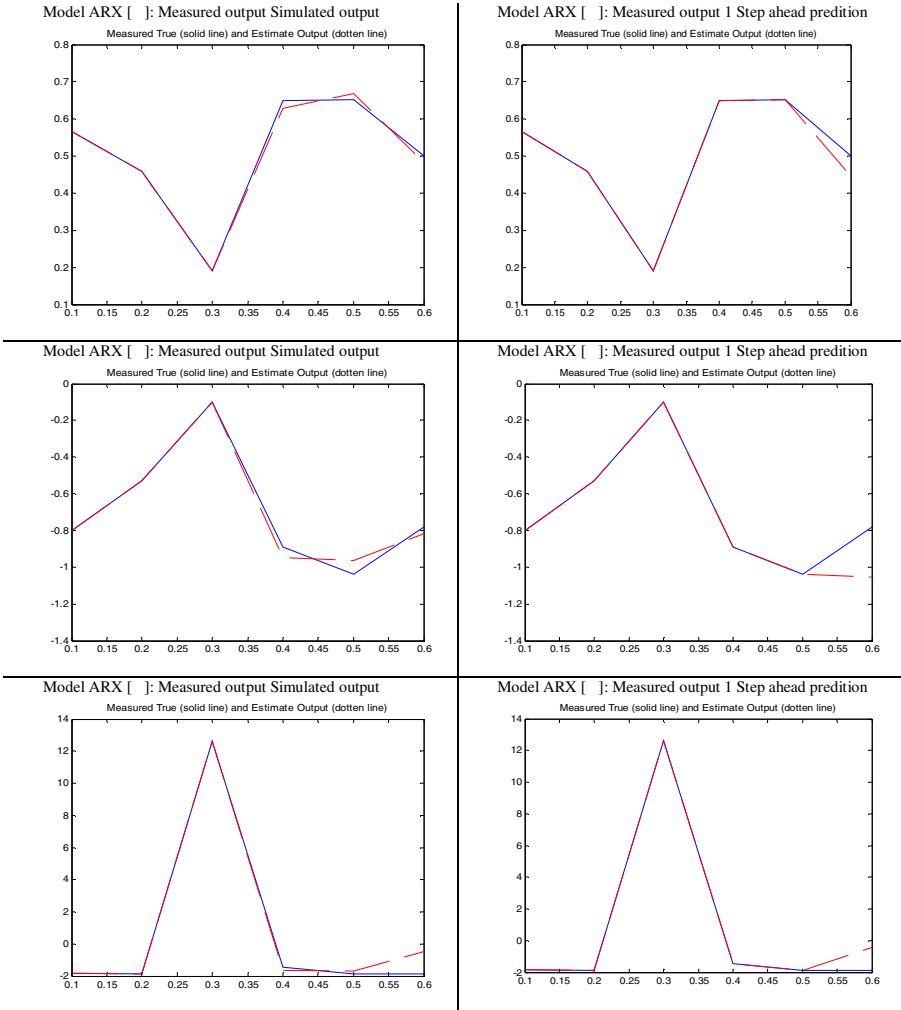


Fig. 2. Representation of measured output, simulated output and one-step-ahead prediction for three black-box models. The model generated by the ARX model for angle error, output $y_1(t)$, is shown in the upper row. On the left, measured output vs. simulated output, on the right, measured output vs. one-step-ahead prediction. The ARX model for the output $y_2(t)$, depth error, is presented in row 2 and finally, the ARX model for output $y_3(t)$, surface roughness is shown in row 3. The validation data set was not used for the estimation of the model. The order of the structure of the model is $[2 \ 1 \ 4 \ 1 \ 1 \ 2 \ 1]$ by model type. The solid line represents true measurements and the dotted line represents estimated output.

It may be seen from Fig. 2 that the ARX model is capable of simulating and predicting the error behaviour of the laser milled piece as it meet the indicators and is capable of modelling more than 90% of the true measurements. This is also evident

Table 3. Indicator values for several proposed models

Indicators and order [na, nb ₁ , nb ₂ , nb ₃ , nk ₁ , nk ₂ , nk ₃]							
		Angle Error		Depth Error		Surface roughness	
Model		[2 1 4 1 1 2 1]	[2 1 4 1 1 3 1]	[2 1 4 1 1 1]	[21 4 1 1 3 1]	[21 4 1 1 2 1]	[21 4 1 1 3 1]
Black- box model, ARX model .	FIT	92.13%	100%	85.98%	100%	89.11%	100%
	FIT1	85.77%	100%	62.96%	100%	88.09%	100%
	FIT10	85.77%	100%	62.96%	100%	88.9%	100%
	V	0.07	0.026	0.0197	0.018	0.249	0.363
	FPE	0.026	0.069	0.051	0.047	0.649	0.088
	NSSE	4.97exp-4	9.32exp-31	0.012	3.01exp-28	0.3549	2.64exp-29

Table 4. Function and parameters that represent the behaviour for angle error of the laser milled piece

Model ARX [2 1 4 1 1 3 1]	
$A(q^{-1})y_1(t) = q^{-n_k} B_1(q^{-1})u_1(t) + q^{-n_k} B_2(q^{-1})u_2(t) + q^{-n_k} B_3(q^{-1})u_3(t) + e(t)$	
Parameters and polynomials.	
$A(q) = 1 - 1.086 q^{-1} + 1.195 q^{-2}$	$B_2(q) = 0.003224 q^{-3} + 0.002786 q^{-4} + 0.000898 q^{-5} + 0.004985 q^{-6}$
$B_1(q) = 0.03113 q^{-1}$	$B_3(q) = 0.01438 q^{-1}$
e(t) is white noise signal with variance 0.119	

Table 5. Function and parameters that represent the behaviour for the depth error of the laser milled piece

Model ARX [2 1 4 1 1 3 1]	
$A(q^{-1})y_1(t) = q^{-n_k} B_1(q^{-1})u_1(t) + q^{-n_k} B_2(q^{-1})u_2(t) + q^{-n_k} B_3(q^{-1})u_3(t) + e(t)$	
Parameters and polynomials.	
$A(q) = 1 - 2.202 q^{-1} + 1.653 q^{-2}$	$B_2(q) = -0.006949 q^{-3} - 0.005614 q^{-4} - 0.002545 q^{-5} - 0.008835 q^{-6}$
$B_1(q) = -0.03203 q^{-1}$	$B_3(q) = -0.03237 q^{-1}$
e(t) represents white noise signal with variance 0.082	

from Table 3. Tables 4, 5, 6 show the function and the parameters that define the laser milling process, on the basis of the ARX model. The tests were performed using Matlab and the System Identification Toolbox.

Table 6. Function and parameters that represent the behaviour for surface roughness of the laser milled piece

Model ARX [2 1 4 1 1 3 1]	
$A(q^{-1})y_1(t) = q^{-n_k} B_1(q^{-1})u_1(t) + q^{-n_k} B_2(q^{-1})u_2(t) + q^{-n_k} B_3(q^{-1})u_3(t) + e(t)$	
Parameters and polynomials.	
$A(q) = 1 + 0.1501 q^{-1} - 0.1302 q^{-2}$	$B_2(q) = -0.004364 q^{-3} - 0.005079 q^{-4} - 0.008746 q^{-5} - 0.005709 q^{-6}$
$B_1(q) = -0.0464 q^{-1}$	$B_3(q) = -0.01484 q^{-1}$
	$e(t)$ is white noise signal with variance 0.153

4 Conclusions and Futures Lines of Work

We have presented an investigation to study and identify the most appropriate modelling system for laser milling of copper components. Several methods were investigated to achieve the best practical solution to this interesting problem. The study shows that the BJ model is best adapted to this case, in terms of identifying the best conditions and predicting future circumstances.

It is important to emphasize that an important aspect of this research lies in the use of a two-phase model when modelling the laser milling process for copper components: a first phase, which applies projection methods to establish whether the data describing the case study is “sufficiently informative”. As a consequence, the first phase eliminates one of the problems associated with these identification systems, which is that of having no prior knowledge of whether the experiment that generated the data group may be considered acceptable and will present sufficient information in order to identify the overall nature of the problem.

Future work will be focus on the study and application of other kinds of materials of industrial interest, such as steel.

Acknowledgments

This work has been made possible thanks to the support received from ASCAMM Centro Tecnológico (<http://www.ascamm.es>), which provided the laser milling data and performed all the laser tests. The authors would especially like to thank Mr. Pol Palouzie and Mr. Javier Diaz for their kind-spirited and useful advice. This research has been partially supported through project BU006A08 of the JCyL.

References

1. Diaconis, P., Freedman, D.: Asymptotics of Graphical Projections. *The Annals of Statistics* 12(3), 793–815 (1984)
2. Corchado, E., Fyfe, C.: Connectionist Techniques for the Identification and Suppression of Interfering Underlying Factors. *Int. Journal of Pattern Recognition and Artificial Intelligence* 17(8), 1447–1466 (2003)

3. Friedman, J.H., Tukey, J.W.: Projection Pursuit Algorithm for Exploratory Data-Analysis. *IEEE Transactions on Computers* 23(9), 881–890 (1974)
4. Corchado, E., MacDonald, D., Fyfe, C.: Maximum and Minimum Likelihood Hebbian Learning for Exploratory Projection Pursuit. *Data Mining and Knowledge Discovery* 8(3), 203–225 (2004)
5. Seung, H.S., Soccia, N.D., Lee, D.: The Rectified Gaussian Distribution. *Advances in Neural Information Processing Systems* 10, 350–356 (1998)
6. Fyfe, C., Corchado, E.: Maximum Likelihood Hebbian Rules. In: *Proc. of the 10th European Symposium on Artificial Neural Networks (ESANN 2002)*, pp. 143–148 (2002)
7. Corchado, E., Han, Y., Fyfe, C.: Structuring Global Responses of Local Filters Using Lateral Connections. *Journal of Experimental & Theoretical Artificial Intelligence* 15(4), 473–487 (2003)
8. Ljung, L.: *System Identification, Theory for the User*. Prentice-Hall, Englewood Cliffs (1999)
9. Nögaard, M., Ravn, O., Poulsen, N.K., Hansen, L.K.: *Neural Networks for Modelling and Control of Dynamic Systems*. Springer, London (2000)
10. Söderström, T., Stoica, P.: *System identification*. Prentice-Hall, Englewood Cliffs (1989)
11. Nelles, O.: *Nonlinear System Identification, From Classical Approaches to Neural Networks and Fuzzy Models*. Springer, Heidelberg (2001)
12. Haber, R., Keviczky, L.: *Nonlinear System Identification, Input-Output Modeling Approach, Part. 2: Nonlinear System structure Identification*. Kluwer Academic Publishers, Dordrecht (1999)
13. Haber, R., Keviczky, L.: *Nonlinear System Identification, Input-Output Modeling Approach, Part 1: Nonlinear System Parameter Estimation*. Kluwer Academic Publishers, Dordrecht (1999)
14. Stoica, P., Söderström, T.: A useful parametrization for optimal experimental design. In: *IEEE Trans. Automatic. Control*, vol. AC-27 (1982)
15. He, X., Asada, H.: A new method for identifying orders of input-output models for nonlinear dynamic systems. In: *Proc. Of the American Control Conf., S.F., California*, pp. 2520–2523 (1993)
16. Akaike, H.: Fitting autoregressive models for prediction. *Ann. Inst. Stat. Math.* 20, 425–439 (1969)
17. Arias, G., Ciurana, J., Planta, X., Crehuet, A.: Analyzing Process Parameters that influence laser machining of hardened steel using Taguchi method. In: *Proceedings of 52nd International Technical Conference SAMPE 2007, Baltimore (2007)*; ISBN 978-0-938994-72-5

Country and Political Risk Analysis of Spanish Multinational Enterprises Using Exploratory Projection Pursuit

Alfredo Jiménez¹, Álvaro Herrero², and Emilio Corchado²

¹Department of Economics and Business Administration
University of Burgos, Spain
alfredojimenezpalmero@hotmail.com

²Department of Civil Engineering, University of Burgos, Spain
C/ Francisco de Vitoria s/n, 09006 Burgos, Spain
{ahcosio, escorchado}@ubu.es

Abstract. As part of a multidisciplinary research project on relevant applications of Exploratory Projection Pursuit, this study sets out to examine levels of country and political risk that are assumed by a sample of Spanish Multinational Enterprises (MNEs). It analyses information pertaining to points such as decisions over the localization of subsidiary firms in various regions across the world, the importance accorded to such decisions and the driving forces behind them. The specific variables under study are economic freedoms, perceived levels of corruption and the constraints affecting the host governments in a sample of 1773 Spanish MNE subsidiaries throughout the world. Several neural projection models are applied, and we are able to conclude that these connectionist techniques help analyse the relevant data to identify the internationalization strategies of Spanish MNEs, their underlying motives and the goals they pursue.

Keywords: multinational firm, country and political risk, foreign direct investment, exploratory projection pursuit, unsupervised learning.

1 Introduction

Internationalization is a decision that firms must take with increasing frequency, as competition in many sectors due to globalization, obliges companies to enter international markets in the search for new markets and lower operating costs. Thus, the decision to become a Multinational Enterprise (MNE) and the challenge of successfully undertaking such a transformation are more relevant than ever. Numerous works have sought to clarify the factors that are involved in decisions concerning the localization of foreign investments, whether between developed countries, from developed countries to developing countries or, to a lesser extent, vice-versa from developing to developed countries.

[1] contains an interesting table with an abundant bibliography of empirical studies concerned with analyzing the importance of certain factors that attract investments. However, it may be seen how the analysis of political risk as a fundamental factor in the localization of direct investment, constitutes a field that has received much less

attention that other factors, with the notable exceptions of Marois [2], [3] for French MNEs, Rich and Mahmoud [4] for Canadian MNEs, Mortanges and Aller [5] for Dutch MNEs, Mutinelli and Picitello [6] for the Italian ones and Noordin et al [7], the last-named being one of the few works centred on multinational firms in a less-developed countries, in this case Malaysia. That is despite some surveys which show that on occasions even 100% of the firms consulted performed assessments of the political risk to which their subsidiaries were exposed [8].

Visualisation techniques have been employed to analyse large datasets for some time. They are considered a viable approach in the search for information and they present it on graphic display devices that highlight different characteristics and allow anomalies to be detected by the relevant decision-makers [9].

The identification of patterns that exist across dimensional boundaries in high dimensional datasets is a challenging task. Such patterns may become visible if changes are made, to the spatial coordinates; however an a priori decision, as to which parameters will reveal most patterns, requires prior knowledge of the unknown patterns.

In this study, EPP models are applied to analyze the internal structures of the aforementioned case study on the role of country and political risk in the localization decisions of Spanish MNEs. The paper is structured as follows. Section 2 outlines the application of dimensionality reduction techniques for data analysis and also describes the main neural projection model applied in this work. Section 3 sets out the dataset on country and political risk, while section 4 presents the results and, finally, Section 5 summarizes the conclusions and the future lines of research.

2 Dimensionality Reduction Visualization for Data Analysis

Projection methods project high-dimensional data points onto lower dimensions in order to identify "interesting" directions in terms of any specific index or projection. Such indexes or projections are, for example, based on the identification of directions that account for the largest variance of a dataset (such as Principal Component Analysis (PCA) [10], [11], [12]) or the identification of higher order statistics such as the skew or kurtosis index, as in the case of Exploratory Projection Pursuit (EPP) [13]. Having identified the interesting projections, the data is then projected onto a lower dimensional subspace plotted in two or three dimensions, which makes it possible to examine its structure with the naked eye. The remaining dimensions are discarded as they mainly relate to a very small percentage of the information or the dataset structure. In that way, the structure identified through a multivariable dataset may be visually analysed with greater ease.

The combination of this type of technique together with the use of scatter plot matrixes constitutes a very useful visualization tool to investigate the intrinsic structure of multidimensional datasets, allowing experts to study the relations between different components, factors or projections, depending on the technique that is used.

2.1 The Unsupervised Connectionist Model

The standard statistical EPP method [13] provides a linear projection of a dataset, but it projects the data onto a set of basic vectors which best reveal the interesting

structure in data; interestingness is usually defined in terms of how far the distribution is from the Gaussian distribution.

One neural implementation of EPP is Maximum-Likelihood Hebbian Learning (MLHL) [14], [15], which identifies interestingness by maximising the probability of the residuals under specific probability density functions that are non-Gaussian.

An extended version of this model is the Cooperative Maximum-Likelihood Hebbian Learning (CMLHL) [16] model. CMLHL, which is based on MLHL [14], [15] adds lateral connections [16], [17] which have been derived from the Rectified Gaussian Distribution [18]. The resultant net can find the independent factors of a data set but does so in a way that captures some type of global ordering in the data set.

Considering an N -dimensional input vector (x), and an M -dimensional output vector (y), with W_{ij} being the weight (linking input j to output i), then CMLHL can be expressed [16], [17] as:

1. Feed-forward step:

$$y_i = \sum_{j=1}^N W_{ij} x_j, \forall i . \quad (1)$$

2. Lateral activation passing:

$$y_i(t+1) = [y_i(t) + \tau(b - Ay)]^+ . \quad (2)$$

3. Feedback step:

$$e_j = x_j - \sum_{i=1}^M W_{ij} y_i, \forall j . \quad (3)$$

4. Weight change:

$$\Delta W_{ij} = \eta \cdot y_i \cdot \text{sign}(e_j) |e_j|^{p-1} . \quad (4)$$

Where: η is the learning rate, τ is the "strength" of the lateral connections, b the bias parameter, p a parameter related to the energy function [14], [15], [16] and A a symmetric matrix used to modify the response to the data [16]. The effect of this matrix is based on the relation between the distances separating the output neurons.

3 Country and Political Risk Dataset

The empirical part of this work seeks to describe the principal characteristics of country and political risk which influence the localization and the presence in foreign countries of Spanish MNEs, while controlling for the effects of variables related both to the firm and to the country.

The sample of firms on which the present study is based is made up of Spanish multinational firms of over 250 employees, which in December 2007 appeared on the list of the *Instituto de Comercio Exterior* [Institute of Foreign Commerce] (ICEX), the www.oficinascomerciales.es web page, and other foreign bodies concerned with foreign direct investment contactable through the ICEX that provide directories of Spanish MNEs with investments in their countries.

In total the sample is formed of 166 Spanish MNEs, which have 1812 subsidiaries localized across the world, for which the data on the necessary variables was obtained for 1773 subsidiaries, which represents 97.7% of all cases. The dataset is composed of 1152 observations relating to 12 variables, namely: Index of Economic Freedom, Corruption Perceptions Index, Political Constraints Index (POLCON), total assets, employee numbers, Return on Equity (ROE), growth rate of sales, solvency ratio, number of foreign countries in which the MNE has its subsidiaries, Foreign Direct Investment/Gross Domestic Product (FDI/GDP), GDP growth as a measure of the appeal of the country, and total population as a measure of size.

4 Experiment, Results and Comparison

CMLHL was applied to the above dataset (See Section 3). The projections obtained by this model are set out below and analyzed in this section.

It can be seen that the CMLHL projection (Fig.1) reflected the different motivations driving Spanish MNEs to localize in the countries of the different regions under analysis, which represent, to a great degree, the main host countries traditionally targeted by Spanish foreign direct investment.

It may be seen that group 1 (Fig. 1) is made up of smaller firms that can not afford to overspend scarce resources on risky internationalization strategies. Something similar happens in group 2, where small firms dedicated to the services sector are concentrated, and in which two subgroups may be seen in accordance with the countries that they were targeting.

Group 3 (Fig. 1) however, refers to firms with quite different characteristics which are, above all, large firms in the manufacturing sector with complex internationalization strategies as a result of the high volume of resources that they are able to invest.

For its part, Group 4 (Fig. 1) contains the densest subgroups in the entire sample. They show a grouping of firms within the currently controversial construction sector that has targeted developed economies with large markets as well as countries that have recently joined the European Union, and which has greater expectations for growth and profitability. However, the existence of subgroups where the presence of Eastern European countries is very predominant, which is the case of subgroup 4.1 (Fig. 1), demonstrates that these countries, despite having achieved important economic, social and institutional progress, still constitute an investment destination with specific characteristics that distinguish them from the other member States of the European community, which is congruent with the results obtained by Durán, de la Fuente and Jiménez (2008b).

Group 5 (Fig. 1) from among all of the subgroups shows that some multinationals seek to minimize the risks associated with investment in countries with close cultural ties, thereby seeking better management and easier solutions to any potential problems that might arise.

Group 6 (Fig. 1) is very interesting as it is made up of the flagships of Spanish foreign investment, and in addition shows a certain short-termism in its investment strategy in Latin America, seeking to achieve a competitive advantage at the start of the investment, but unconcerned about its inherent problems that relate to political risk.

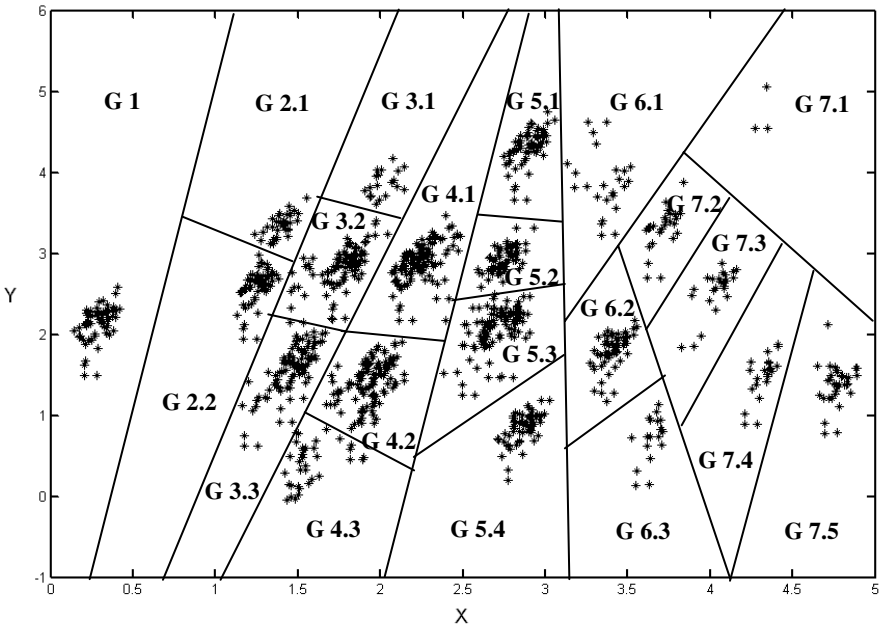


Fig. 1. CMLHL projection for the data set

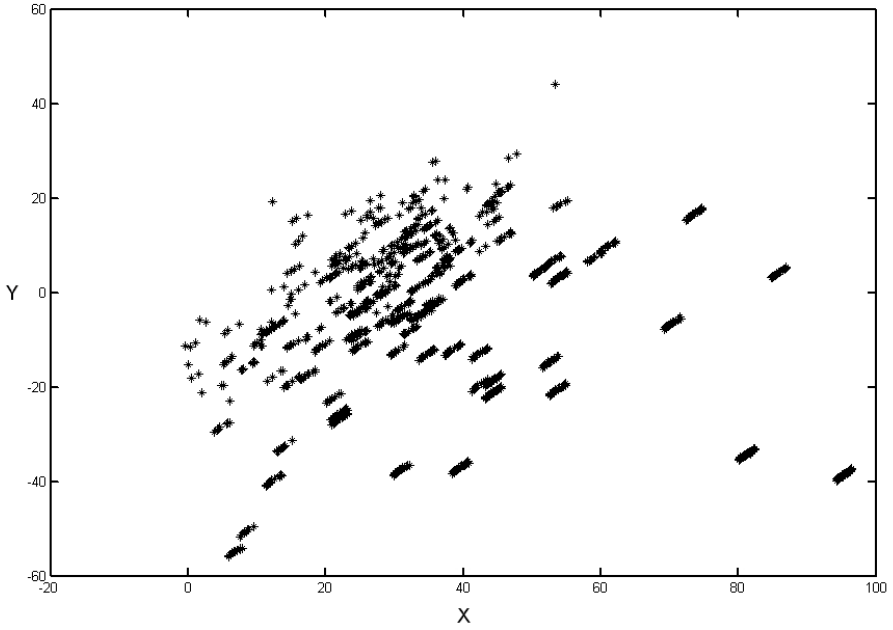


Fig. 2. PCA projection for the data set

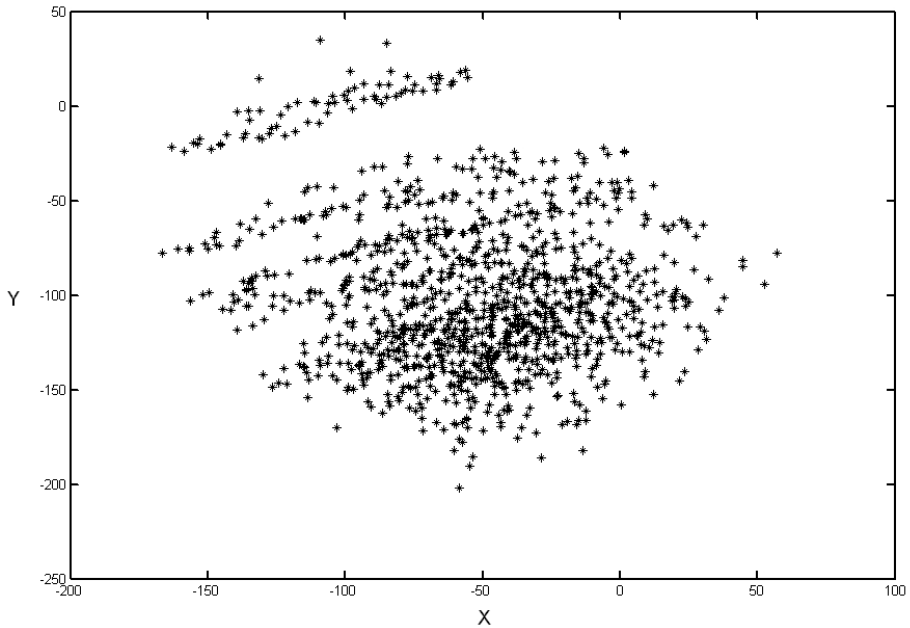


Fig. 3. CCA projection for the data set

Finally, Group 7 (Fig. 1) constitutes a set of localizations with the highest number of subgroups, and in all of them the high dispersion between their constituent elements may easily be appreciated. Those firms with the least clear internationalization strategies appear here, and at times are similar to those of other groups, but in different countries or other sectors.

Some other well-known projection models, namely Principal Component Analysis (PCA) [10], [11], [12] and Curvilinear Component Analysis (CCA) [19] were applied to confirm the validity of the results. Figs. 2 and 3 show the projections obtained by these models.

Fig 2. presents the projection obtained by PCA through the two first principal components. As it can be seen, PCA is able to show the structure of the dataset but in a less clear way than CMLHL (Fig. 1).

Fig 3. presents the projection obtained CCA. In this case, CCA is not able to show the inner structure of the data, as only 2 main groups could be differentiated.

CMLHL (Fig. 1) clearly obtains a clearer and more widely spread projection than both PCA (Fig. 2) and CCA (Fig. 3), allowing more visual information to be extracted, which in turn enables clearer and better conclusions to be drawn from the data.

5 Conclusions and Future Work

In brief, we can conclude that this study has served to show the different reasons underlying the internationalization strategies of Spanish MNEs and the different goals

they pursue, which may be appreciated from the different groups identified by CMLHL, localizing in a specific country according to their specific needs or those of their sector, as is evinced by the different subgroups.

Future work will focus on the study of more international areas and also of international companies other than the Spanish ones. Also other unsupervised neural models such as topology preserving maps will be applied, for comparison purposes, to this interesting case study.

Acknowledgments. This research has been partially funded through project BU006A08 of the JCyL.

References

1. Galan, J.I., Gonzalez-Benito, J., Zuniga-Vincente, J.A.: Factors Determining the Location Decisions of Spanish MNEs: An Analysis based on the Investment Development Path. *J. Int. Bus. Stud.* 38(6), 975–997 (2007)
2. Marois, B.: Assessment and Management of Political Risk: Practice of French Firms. In: Annual Meeting of the Academy of International Business, London (1979)
3. Marois, B.: Comment les Entreprises Francaises Gerent le Risque Politique. *Revue Francaise de Gestion*, 4–9 (May–August 1981)
4. Rich, G., Mahmoud, E.: Political Risk Forecasting by Canadian Firms. *International Journal of Forecasting* 6(1), 89–102 (1990)
5. Mortanges, C.P.d., Allers, V.: Political Risk Assessment: Theory and the Experience of Dutch Firms. *International Business Review* 5(3), 303–318 (1996)
6. Mutinelli, M., Piscitello, L.: Differences in the Strategic Orientation of Italian MNEs in Central and Eastern Europe. The Influence of Firm-specific Factors. *International Business Review* 6(2), 185–205 (1997)
7. Noordin, B.A., Harjito, D.A., Hazir, A.Y.: Political Risk Assessment of Malaysian based Multinational Corporation. *Problems and Perspectives in Management* 4(3), 91–99 (2006)
8. Hashmi, M.A., Guvenli, T.: Importance of Political Risk Assessment Function in U.S. Multinational Corporations. *Global Finance Journal* 3(2), 137–144 (1992)
9. Ahlberg, C., Shneiderman, B.: Visual Information Seeking: Tight Coupling of Dynamic Query Filters with Starfield Displays. In: Proceedings of the SIGCHI conference on Human Factors in Computing Systems. ACM, New York (1994)
10. Hotelling, H.: Analysis of a Complex of Statistical Variables Into Principal Components. *Journal of Education Psychology* 24, 417–444 (1933)
11. Pearson, K.: On Lines and Planes of Closest Fit to Systems of Points in Space. *Philosophical Magazine* 2(6), 559–572 (1901)
12. Oja, E.: Neural networks, principal components, and subspaces. *Int. Journal of Neural Systems* 1, 61–68 (1989)
13. Friedman, J.H., Tukey, J.W.: A Projection Pursuit Algorithm for Exploratory Data-Analysis. *IEEE Transactions on Computers* 23(9), 881–890 (1974)
14. Corchado, E., MacDonald, D., Fyfe, C.: Maximum and Minimum Likelihood Hebbian Learning for Exploratory Projection Pursuit. *Data Mining and Knowledge Discovery* 8(3), 203–225 (2004)
15. Fyfe, C., Corchado, E.: Maximum Likelihood Hebbian Rules. In: Proc. of the 10th European Symposium on Artificial Neural Networks (ESANN 2002), pp. 143–148 (2002)

16. Corchado, E., Fyfe, C.: Connectionist Techniques for the Identification and Suppression of Interfering Underlying Factors. *Int. Journal of Pattern Recognition and Artificial Intelligence* 17(8), 1447–1466 (2003)
17. Corchado, E., Han, Y., Fyfe, C.: Structuring Global Responses of Local Filters Using Lateral Connections. *Journal of Experimental & Theoretical Artificial Intelligence* 15(4), 473–487 (2003)
18. Seung, H.S., Socoli, N.D., Lee, D.: The Rectified Gaussian Distribution. *Advances in Neural Information Processing Systems* 10, 350–356 (1998)
19. Demartines, P., Hérault, J.: Curvilinear Component Analysis: A Self-Organizing Neural Network for Nonlinear Mapping of Data Sets. *IEEE Transactions on Neural Networks* 8(1), 148–154 (1997)

Single-Layer Neural Net Competes with Multi-layer Neural Net

Zheng Rong Yang

School of Biosciences, University of Exeter, UK

Abstract. This paper presents a novel neural network with only one layer which can compete with multi-layer neural nets. This novel neural net is called a double-threshold single-layer neural net. The theoretical analysis and experiments show that it can demonstrate similar performance as multi-layer neural nets.

Keywords: neural networks, multi-layer feed-forward, sigmoid function.

1 Introduction

Neural networks or multi-layer neural nets (MLNs) are a class of machine learning algorithms which are capable of modeling experimental data when domain knowledge about an exact model format is incomplete or unknown. Since the development [1], MLNs have been widely used in many applications. The basic principle of MLNs is the parallelism and the mathematical background. The parallelism lies as the fact that the computation of the neurons at the same layer can be independent from each other. If we treat the input layer, the hidden layer and the output layer as three nodes in a Markov Chain model, where the computation at each layer except for the input layer is dependent on the computation at its subsequent lower layer. MLNs also have the mathematical background in numerical computation where various optimization techniques can be directly employed. There are mainly two types of optimization procedures for MLN model construction, namely the update rule relying on the first derivatives and the update rule based on the second derivatives. Either has a different learning performance for different applications [2].

The most commonly used learning rule (or update rule) is based on the first derivative [3]. The other learning rules include the weight decay [4], the introduction of the Hessian (second derivatives) [5], Quickprop method [6], conjugate method [7], and pruning method [8]. In order to address the learning efficiency of MLN, a number of new algorithms have been proposed. For instance, the second order convergence rate is analyzed leading to a fast calculation of the Hessian matrix for speeding neural network learning [9]. The network weights can also be estimated by minimizing the traditional mean square error function, where a global heuristic search uses a so-called LP τ strategy based on the search of low-discrepancy sequences of points plus a simplex local search [10].

It must be noted that the powerfulness of MLNs is the use of the hidden neurons which helps deal with nonlinearity. In fact, a MLN is a well-organized team of single-layer neural nets (SLNs). For instance, the left panel of Fig. 1 shows a common

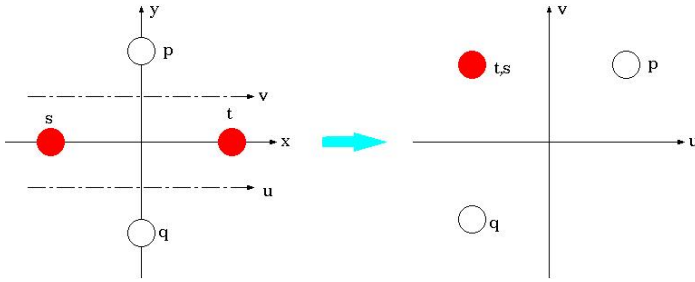


Fig. 1. XOR data to show how the original data space which is not linearly separable is mapped to a data space which is linearly separable. Two input variables are x and y . The new space has two dimensions, i.e. u and v . Four data points are labeled as p , q , s , and t .

nonlinear classification data (XOR data) in two dimensions with four data points belonging to two classes. In this data space, no linear classification algorithm will be useful. However, if we insert two lines (z_1 and z_2) and then map these four data points to these two lines. Another data space is formed as shown in the right panel of Fig. 1. This new data space is linearly separable and a linear classification algorithm can be used for a perfect classification between two classes. Treating z_1 and z_2 as two SLNs and adding one more SLN on top of them, a MLN with two hidden neurons is formed. It has been proven that a MLN with two hidden neurons can handle the XOR data efficiently.

However, the use of the hidden neurons brings problems in computation. Training a MLN in an application is not an easy task. In most cases, we may not have a clear idea about the right number of hidden neurons. Many trial-and-error times are therefore needed for finding a right model for an application. The CPU time are normally heavily used for optimizing MLN parameters which is also not a trivial task. Removing hidden neurons certainly brings us back to SLN which is not useful for nonlinear problems.

This paper proposes a novel neural network structure called double-threshold single-layer neural net (dSLN). The core principle of dSLN is to introduce a double-threshold sigmoid function by which the network is able to handle nonlinear data.

2 SLN and MLN

Let $\mathbf{x}_n \in \mathfrak{R}^D$ be an input vector and $y_n \in (0,1)$ be the corresponding prediction of a model. In a commonly used single-layer neural net (SLN), the model output occurred at the output neuron is normally defined by a sigmoid function ($\sigma(\circ)$) which has a single threshold (zero) within the exponential function, $\sigma(z_n) = [1 + \exp(-z_n)]^{-1}$ or $\sigma(\mathbf{x}_n, \mathbf{w}) = [1 + \exp(-\mathbf{x}_n \cdot \mathbf{w})]^{-1}$ with $z_n = \mathbf{x}_n \cdot \mathbf{w}$ and $\mathbf{w} \in \mathfrak{R}^D$ is a weight vector of SLN. It can be further generalized by introducing two parameters for weighting and biasing the component of the sigmoid function

$$y_n = \sigma(\alpha, \beta, \mathbf{x}_n, \mathbf{w}) = \frac{1}{1 + \exp(-(\alpha \mathbf{x}_n \cdot \mathbf{w} - \beta))} \tag{1}$$

$\alpha > 0$ is called a scaling parameter and β is called a location parameter acting as the threshold. The scaling parameter is related with the decision surface. A large value of α makes a small decision margin while a small α value makes a large decision margin. The location parameter is related with the decision boundary.

It can be seen that if $\alpha \mathbf{x}_n \cdot \mathbf{w} > \beta$ $y_n > 0.5$, otherwise $y_n < 0.5$. Using this sigmoid function, a space is naturally divided into two halves in a classification problem, one being $\alpha \mathbf{x}_n \cdot \mathbf{w} > \beta$ and the other $\alpha \mathbf{x}_n \cdot \mathbf{w} < \beta$. If this SLN is used for classification analysis, it can be seen that it can only handle linearly separable data. In MLN, hidden neurons are introduced. For simplicity, the scaling parameter and the location parameter are dropped, but they are easily added in. The model output is defined as $y_n = [1 + \exp(-\mathbf{r}_n \cdot \mathbf{w}_o)]^{-1}$ and $r_{nh} = [1 + \exp(-\mathbf{x}_n \cdot \mathbf{w}_h)]^{-1}$. Here $\mathbf{w}_h \in \mathfrak{R}^D$ is the weight vector connecting the hth hidden neuron and input variables, $\mathbf{w}_o \in \mathfrak{R}^H$ is the output neuron weight vector, and $\mathbf{r}_n \in (0,1)^H$ is the hidden neuron output vector.

3 Double-Threshold SLN

We now look at how to revise the existing sigmoid function so as to develop a new SLN for handling nonlinear data. In equation (1), it can be seen that a sigmoid function is shaped by two important parameters, the scaling and the location parameters. The larger the scaling parameter, the sharper the decision surface is. The location parameter is used to determine where we will place a boundary for separating two classes. Because the sigmoid function defined in equation (1) has a typical property that the function output is monotonically proportional to the argument of the function, i.e. $\sigma(x) \propto x$. The linearity lies here. Suppose another sigmoid function is designed as below,

$$\sigma(\alpha^-, \beta^-, x) = \frac{1}{1 + \exp(\alpha^- x - \beta^-)} \tag{2}$$

where $\alpha^- > 0$. It can be seen that $\sigma(x) \propto \frac{1}{x}$. The sigmoid function defined in equation (1) is referred to as a positive sigmoid function as is denoted as $\sigma(\alpha^+, \beta^+, x)$ or simply $\sigma^+(x)$ while the sigmoid function defined in equation (2) is referred to as a negative sigmoid function (simply $\sigma^-(x)$). Multiplying the positive sigmoid function by the negative sigmoid function leads to $y(x) = \sigma^+(x)\sigma^-(x)$. This novel function is illustrated in Fig. 2, where it is supposed that the following relations hold $\alpha^- < 0$, $\alpha^+ > 0$, and $\beta^- > \beta^+$. From this combination, it is expected that the model output

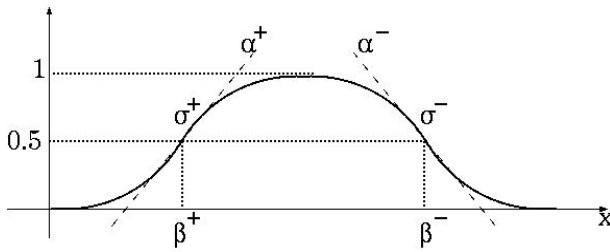


Fig. 2. Double sigmoid functions as model output

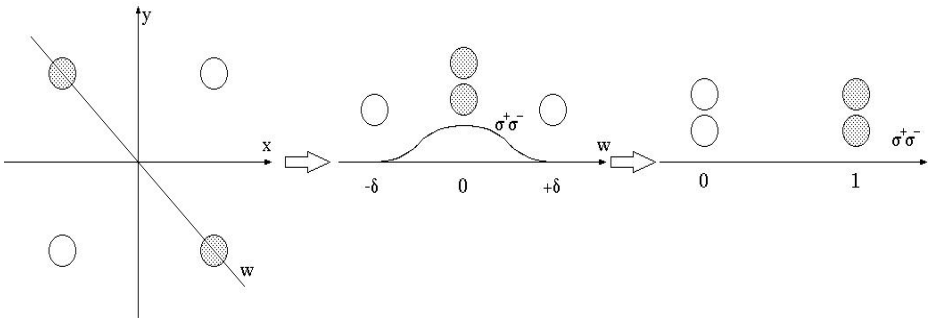


Fig. 3. The handling of the XOR data using the double sigmoid function

will no longer be proportional to the input variable universally, i.e. it is impossible to find a relationship like $y \propto x$ universally. The relation $y \propto x$ holds in the interval $(-\infty, (\alpha^+ \beta^+ + \alpha^- \beta^-) / (\alpha^+ + \alpha^-))$ and the relation $y \propto \frac{1}{x}$ holds in the interval $[(\alpha^+ \beta^+ + \alpha^- \beta^-) / (\alpha^+ + \alpha^-), +\infty)$. This property provides a possibility to handle nonlinear data.

Fig. 3 shows such a possibility to handle the XOR data. A neural net employing this double-sigmoid function structure is referred to as the double-threshold single-layer neural net (dSLN), where β^+ and β^- are two thresholds (boundaries).

4 Use Double-Threshold SLN for Classification Analysis

In a classification analysis task, we normally have a cross-entropy function to quantify the likelihood of a model, i.e. $L = \prod_{n=1}^N y_n^{t_n} (1 - y_n)^{1-t_n}$. After applying the negative logarithm and adding a regularization term, the objective function is then designed as below

$$O = -\sum_{n=1}^N t_n \log(y_n) + (1 - t_n) \log(1 - y_n) + \frac{1}{2} \lambda \theta^T \theta \tag{3}$$

Here $\lambda > 0$ is called a regularization constant and $\theta = w \cup \{\alpha^+, \beta^+, \alpha^-, \beta^-\}$. For simplicity, both scaling parameters have been omitted and are left for space limit of this paper. The first derivative of the objective function of O with respect to w_d is

$$\nabla O(w_d) = \lambda w_d + \sum_{n=1}^N \frac{\sigma_n^- - \sigma_n^+}{1 - y_n} e_n x_{nd} \tag{4}$$

In equation (4), the fraction needs to be checked if it will cause numerical problem.

Theorem 1. $\sigma_n^- - \sigma_n^+$ converges to zero faster than $1 - y_n$ when $y_n \rightarrow 1$.

Prove. First, three quantities (σ_n^- , σ_n^+ , and y_n) are treated as functions of $z_n = \mathbf{x}_n \cdot \mathbf{w}$. The subscript n is also dropped. Therefore, the fraction in equation (4)

can be written as $\frac{\sigma^-(z) - \sigma^+(z)}{1 - y(z)}$. Because $\sigma^-(z) \in (0,1)$ and $\sigma^+(z) \in (0,1)$,

$\sigma^-(z)\sigma^+(z) < \sigma^+(z)$. This leads to $\sigma^-(z)\sigma^+(z) - \sigma^-(z) < \sigma^+(z) - \sigma^-(z)$

and $\sigma^-(z) - \sigma^-(z)\sigma^+(z) > \sigma^-(z) - \sigma^+(z)$. Because $\sigma^-(z) < 1$, $1 < \frac{1}{\sigma^-(z)}$ and

$$1 - \sigma^+(z) < \frac{1}{\sigma^-(z)} - \sigma^+(z) \tag{5}$$

or $\sigma^-(z) - \sigma^+(z) < \sigma^-(z) - \sigma^-(z)\sigma^+(z) < 1 - \sigma^-(z)\sigma^+(z)$. Finally, the following equation holds

$$\sigma^-(z) - \sigma^+(z) < 1 - y(z) \text{ or } \frac{\sigma^-(z) - \sigma^+(z)}{1 - y(z)} < 1 \tag{6}$$

Equation (6) shows that $\sigma^-(z) - \sigma^+(z)$ converges to zero faster than $1 - y(z)$ for any value of z or

$$\lim_{z \rightarrow z_0} \frac{\sigma^-(z) - \sigma^+(z)}{1 - y(z)} < \infty \tag{7}$$

Here z_0 is the point where $y(z_0) \rightarrow 1$ in a classification model. The above theorem indicates that for $|z - z_0| < \varepsilon$, it is almost true that $1 - y(z) < \delta(\varepsilon)$ can hold and $\sigma^-(z) - \sigma^+(z) \ll \delta(\varepsilon)$. Based on the above analysis, the update rule for \mathbf{w} can be safely implemented as below

$$\Delta \mathbf{w} = -\eta (\lambda \mathbf{w} + \mathbf{X}^T \mathbf{B} \mathbf{e}) \tag{8}$$

where $\eta > 0$ is a positive value for small incremental move, $\mathbf{e} = (e_1, e_2, \dots, e_N)$ is the error vector, $\mathbf{B} = \text{diag}\left\{\frac{\sigma_n^- - \sigma_n^+}{1 - y_n}\right\}$ is a diagonal matrix, and \mathbf{X} is the input matrix. The first derivative of O with respect to β^+

$$\nabla O(\beta^+) = \lambda \beta^+ - \sum_{n=1}^N \frac{1 - \sigma_n^+}{1 - y_n} e_n \tag{9}$$

In equation (9), there is another fraction which needs to be proved without any risk of numerical collapse.

Theorem 2. $1 - \sigma_n^+$ converges to zero faster than $1 - y_n$.

Prove. We use the same treatment used in proving the theorem 1. Because $\sigma^-(z) \in (0,1)$ and $\sigma^+(z) \in (0,1)$, $y(z) = \sigma^-(z)\sigma^+(z) < \sigma^+(z)$ or $1 - \sigma^+(z) < 1 - y(z)$. This shows that

$$\lim_{z \rightarrow z_0} \frac{1 - \sigma^+(z)}{1 - y(z)} < \infty \tag{10}$$

The update rule for β^+ is then

$$\Delta \beta^+ = -\eta \left(\lambda \beta^+ - \sum_{n=1}^N \frac{1 - \sigma_n^+}{1 - y_n} e_n \right) \tag{11}$$

The first derivative of O with respect to β^- is

$$\nabla O(\beta^-) = \lambda \beta^- - \sum_{n=1}^N \frac{1 - \sigma_n^-}{1 - y_n} e_n \tag{12}$$

The update rule for β^- is

$$\Delta \beta^- = -\eta \left(\lambda \beta^- - \sum_{n=1}^N \frac{1 - \sigma_n^-}{1 - y_n} e_n \right) \tag{13}$$

5 Use Double-Threshold SLN for Regression Analysis

In a regression model, $t_n \in [0,1]$ and the regularized least square error function is defined as

$$O = \frac{1}{2} \sum_{n=1}^N (t_n - y_n)^2 + \frac{1}{2} \lambda \boldsymbol{\theta}^T \boldsymbol{\theta} \tag{14}$$

The update rule for \mathbf{w} is $\Delta \mathbf{w} = -\eta (\lambda \mathbf{w} - \mathbf{X}^T \Lambda \mathbf{e})$. Here $\Lambda = \text{diag}\{y_n(1 - y_n)\}$ is the entropy matrix. The update rule for β^+ is $\Delta \beta^+ = -\eta \left(\lambda \beta^+ + \sum_{n=1}^N (1 - \sigma_n^+) y_n e_n \right)$. The update rule for β^- is $\Delta \beta^- = -\eta \left(\lambda \beta^- - \sum_{n=1}^N (1 - \sigma_n^-) y_n e_n \right)$.

6 Results on the XOR Data

For the XOR data, SLN did not work as expected while both MLN and dSLN work well with 100% accuracy. For dSLN, $\beta^+ = 3.5$ and $\beta^- = 3.5$.

7 Results on the Toy Regression Data

Table 1 shows the simulation on a sine data set using SLN, MLN with 10, 100 and 500 hidden neurons as well as dSLN, where two zones (two pairs of positive and negative sigmoid functions) are used for dSLN. Fig. 4 shows the simulation results using MLNs with 100 hidden neurons as well as dSLN. From Table 1 as Fig. 4, we can see that dSLN works comparably with MLN. For dSLN, two sets of sigmoid function parameters are $(\beta_1^+ = -3.39, \beta_1^- = -1.77)$ and $(\beta_2^+ = 1.95, \beta_2^- = 3.38)$.

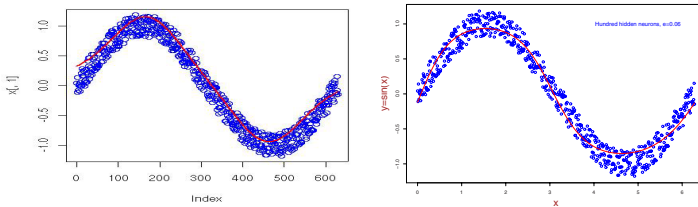


Fig. 4. The simulations on the sin data, where dots are the original data points and curves are the approximated sin function

Table 1. The comparison of dSLN with SLN and MLN on the sine data set

Algorithm	error	
SLN	0.59	
MLN(10)	0.40	10 hidden neurons
MLN(100)	0.07	100 hidden neurons
MLN(500)	0.17	500 hidden neurons
dSLN(2)	0.01	2 pairs of double-sigmoid functions

8 Results on Two Benchmark Classification Data Sets

Two data sets are used for the comparison. They are the Iris data and the wine data. Both have been wide used for developing various machine learning algorithms. In this comparison, SLN and MLN with various hidden neurons are used. Table 2 shows the results on the Iris data which have three classes. The comparison shows that dSLN is comparable with MLN with various hidden neurons with a slightly improved accuracy.

Table 2. The comparison of dSLN with SLN and MLN on the Iris data set. Sp means specificity, Sn means sensitivity, Tot means total accuracy. MCC means the Mathewer correlation coefficient. AUR means the area under ROC curve.

Algorithm	Class	Sp	Sn	Tot	MCC	AUR
SLN	1	1.00	1.00	1.00	1.00	1.00
SLN	2	0.56	0.66	0.59	0.20	0.60
SLN	3	0.97	0.92	0.95	0.89	0.99
MLN(2)	1	1.00	1.00	1.00	1.00	1.00
MLN(2)	2	0.81	1.00	0.87	0.76	0.95
MLN(2)	3	0.97	0.92	0.95	0.89	0.99
MLN(5)	1	1.00	1.00	1.00	1.00	1.00
MLN(5)	2	0.96	0.92	0.95	0.88	0.99
MLN(5)	3	0.97	0.92	0.95	0.89	0.99
MLN(10)	1	1.00	1.00	1.00	1.00	1.00
MLN(10)	2	0.96	0.95	0.97	0.88	0.99
MLN(10)	3	0.97	0.92	0.95	0.89	0.99
dSLN	1	1.00	0.98	0.99	0.99	1.00
dSLN	2	0.96	0.94	0.95	0.90	0.99
dSLN	3	0.97	0.96	0.97	0.93	0.99

Table 3. The comparison of dSLN with SLN and MLN on the wine data set. Sp means specificity, Sn means sensitivity, Tot means total accuracy. MCC means the Mathewer correlation coefficient. AUR means the area under ROC curve.

Algorithm	Class	Sp	Sn	Tot	MCC	AUR
SLN	1	0.97	1.00	0.98	0.96	1.00
SLN	2	0.97	1.00	0.98	0.96	1.00
SLN	3	0.97	1.00	0.98	0.96	1.00
MLN(2)	1	0.99	1.00	0.99	0.99	1.00
MLN(2)	2	0.98	0.98	0.98	0.97	0.99
MLN(2)	3	0.98	1.00	0.99	0.99	0.99
MLN(5)	1	1.00	0.99	0.99	0.99	1.00
MLN(5)	2	0.97	0.98	0.97	0.95	0.99
MLN(5)	3	0.98	1.00	0.99	0.97	0.99
MLN(10)	1	0.99	1.00	0.99	0.98	1.00
MLN(10)	2	0.97	0.98	0.97	0.95	0.99
MLN(10)	3	0.98	1.00	0.99	0.97	0.99
dSLN	1	0.96	1.00	0.97	0.94	1.00
dSLN	2	0.96	1.00	0.97	0.94	1.00
dSLN	3	0.96	1.00	0.97	0.94	1.00

Table 3 shows the comparison on the wine data where there are also three classes. It can be seen that three algorithms are comparable. It demonstrated that dSLN will not offer any benefit if data are linearly separable.

9 Conclusion

This paper has developed a novel neural learning algorithm called the double-threshold single-layer neural net. It aims to maintain the capability of single-layer neural net but explore the power of handling nonlinear data. The algorithm has been applied to two toy data sets, one being classification and one being regression data showing the power of the algorithm. The algorithm has also been applied two typical benchmark data, the Iris data set and the wine data set. The former shows some nonlinearity for class 2 while the second data set shows little nonlinearity. The comparison shows that dSLN outperforms SLN and is comparable with MLN for handling nonlinear data as expected.

References

1. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning internal representations by back-propagating errors. *Nature* 323, 533–536 (1986)
2. Bishop, C.M.: *Neural Networks and Pattern Recognition*. Oxford Press, London (1995)
3. Duda, R.O., Hart, P.E., Stork, D.G.: *Pattern Classification*. John Wiley & Sons, Inc., New York (2002)
4. Cun, Y.L.: A learning scheme for asymmetric threshold networks. In: *Proceedings of Cognitive, Paris, France*, vol. 85, pp. 599–604 (1985)
5. Hassibi, B., Stork, D.G., Wolff, G., Watanabe, T.: Optimal brain surgeon: extensions and performance comparison. In: Cowan, J.D., Tesauro, G., Alspector, J. (eds.) *Advances in Neural Information Processing Systems*, vol. 6, pp. 263–270. Morgan Kaufmann, San Mateo (1994)
6. Fahlman, S.E.: Faster-learning variations on back-propagation: An empirical study. In: Sejnowski, T.J., Hinton, G.E., Touretzky, D.S. (eds.) *1988 Connectionist Models Summer School*, San Mateo, CA, Morgan Kaufmann, San Francisco (1988)
7. Lawrence, S., Giles, C.L.: Overfitting and Neural Networks: Conjugate Gradient and Backpropagation. In: *IEEE-INNS-ENNS International Joint Conference on Neural Networks*, vol. 1, p. 1114 (2000)
8. Reed, R.: Pruning algorithms – a survey. *IEEE Trans. on Neural Networks* 4, 740–747 (1993)
9. Jordanov, I., Georgieva, A.: Neural network learning with global heuristic search. *IEEE Trans. on Neural Networks* 18, 937–942 (2007)
10. Peng, J.X., Li, K., Irwin, G.W.: A new Jacobian matrix for optimal learning of single-layer neural networks. *IEEE Trans. on Neural Networks* 19, 119–129 (2008)

Semi-supervised Growing Neural Gas for Face Recognition

Shireen Mohd Zaki and Hujun Yin

School of Electrical and Electronic Engineering
The University of Manchester
Manchester, M60 1QD, United Kingdom
Shireen.Mohd-zaki@student.manchester.ac.uk,
h.yin@manchester.ac.uk

Abstract. In many face recognition and other classification applications, there exist unlabelled data available for training along with labelled data. The use of unlabelled data can improve the performance of the classifier. In this paper, a semi-supervised growing neural gas is proposed for such applications. The classifier is first trained on the labelled data and then gradually unlabelled data is classified and added to the training data. The proposed algorithm is demonstrated, on both artificial and real datasets, to significantly boost the classification rate with the use of unlabelled data. The improvement is particularly great when the labelled dataset is small. The algorithm is computationally simple and easy to implement.

Keywords: Growing neural gas, classifier, semi-supervised learning. Face recognition.

1 Introduction

In many pattern recognition and classification problems, a classifier is built on a set of training samples, which often are pre-labelled or pre-classified examples. With ever increasing amount of data made available, the process to acquire labelled samples or manually classify or annotate samples is becoming increasingly difficult, expensive or even impractical. Therefore the use of unlabelled data along with labelled data has attracted a great deal of attention recently; and many studies have shown that unlabeled data, when used in conjunction with labelled data, can indeed produce considerable improvement in learning accuracy. Such techniques are often referred to as semi-supervised learning. Such application has been treated in the past as missing value problems and tackled by using mainly unsupervised learning such as the expectation and maximisation algorithm. Semi-supervised learning offers a new approach to the problem and has great potential in data-driven applications.

In this paper, the growing neural gas (GNG) is first described as the basis of constructing a classifier. Then a combined use of labelled and unlabelled data for training a GNG classifier in a semi-supervised fashion is proposed to boost the classification performance, followed by two illustrative experiments and artificial dataset and face recognition application.

2 Growing Neural Gas

Growing neural gas (GNG) is an unsupervised incremental clustering and classification algorithm proposed by Bernd Fritzke [1], developed and extended from the widely used unsupervised clustering algorithms, self-organising maps (SOM) of Teuvo Kohonen [2] and the neural gas (NG) of Thomas Martinetz and Klaus Schulten [3]. GNG method is different from the previous algorithms where the parameters do not change over time as opposed to for example, learning rate in SOM decreases at every learning step. As GNG is an incremental algorithm, there is no need to determine *a priori* the number of nodes. The shape and the size of the network are determined during the simulation, while SOM and NG often train a fixed network throughout. A predetermined network structure has been noted to have limitations on the resulting mappings.

GNG is a combination of earlier work of Bernd Fritzke's growing cell structures (GCS) [4] and Martinetz and Schulten's competitive Hebbian learning (CHL) [5]. The network topology of GNG is generated incrementally by CHL algorithm which successively inserts topological connections or *edge*. The main principle of CHL is: *For each input signal x connect the two closest centres (measured by Euclidean distance) by an edge.*

The growth mechanism of GCS is the main idea of GNG, where nodes are inserted into an initially small network by evaluating local statistical measures gathered during previous adaptation steps. Then some nodes will be removed when there are no more edges emanating from the node that have been removed by their local edging age. The GNG algorithm assumes that every node i , consists of a reference vector, $w_i \in \mathbf{R}^n$, a local accumulated error variable, e_i , and a set of edges representing the topological neighbours of node i . The complete procedure of the GNG algorithm is presented as follows [1];

- (1) Start with two units i and j at random position in the input space.
- (2) Present input vector x from the input set or according to input distribution.
- (3) Find nearest unit s_1 and the second nearest unit s_2 .
- (4) Increment the age of all edges emanating from s_1 .
- (5) Update local error variable by adding the squared distance between w_{s_1} and x :

$$\Delta error(s_1) = \|w_{s_1} - x\|^2$$

- (6) Move s_1 and all its topological neighbours (i.e. all the nodes connected to s_1 by an edge) towards x by fractions of e_b and e_n of the distance:

$$\Delta w_{s_1} = e_b(x - w_{s_1})$$

$$\Delta w_n = e_n(x - w_n) \text{ for all direct neighbours of } s_1$$

- (7) If s_1 and s_2 are connected by an edge, set the age of the edge to 0 (refresh). If there is no such edge, create one.
- (8) Remove edges with age larger than a_{max} . If this result in w has no emanating edges, remove them as well.

(9) If the number of input vectors presented or generated so far is an integer of multiple of a parameter λ , insertion of a new node r as follows:

- Determine unit q with the largest error.
- Among neighbours of q , find node f with the largest error.
- Insert new node r halfway between q and f as follows:

$$w_r = \frac{w_q + w_f}{2}$$

- Create edges between r and q , and r and f . Remove edge between q and f .
- Decrease error variable of q and f by multiplying them with a constant α . Set error r with new error variable of q .

(10) Decrease all error variables of all nodes i by a factor of β .

(11) If the stopping criterion is not met, go back to step (1). For our experiments, the stopping criterion has been set to be the maximum network size.

In our experiments, GNG parameters are used to ensure an unbiased result. The details of GNG parameters used in this experiment are as follows:

- Parameter $\lambda = 300$
- Adaptation parameter for best matching unit $e_b = 0.2$
- Adaptation parameter for neighbouring nodes $e_n = 0.006$
- Decrease parameter of $\alpha = 0.5$
- Decrease parameter of $\beta = 0.995$

3 Proposed Semi-supervised GNG for Face Recognition

There have been growing interests in combining labelled and unlabelled data to improve the performance of clustering and classification. This has been driven by the fact that in many practical applications readily labelled data are hard and expensive to obtain. Unlabelled data has been noted in various researches to assist the labelled data in improving the performance of a classifier, for example as in [6] [7] and [8]. This paper proposes a semi-supervised learning to growing neural gas for face recognition and classification system. The proposed semi-supervised learning is a two-stage method where labelled data are used to train a classifier first and then unlabelled data are labelled according to the trained classifier from the originally labelled data. The second stage involves classifying unlabelled data and re-training the classifier from the classified unlabelled data as well as the originally labelled data to boost the classification rate.

In this paper, a classifier is trained with the GNG algorithm using labelled samples only and unlabelled samples are presented iteratively and labelled one point at a time. Then the newly labelled samples are added into the originally labelled data pool and this process continues until all the unlabelled data are labelled. This new set of labelled data is then being presented to the GNG based classifier and the classification performance are evaluated again. This iterative labelling is termed semi-supervised growing neural gas (SSGNG) and can be described in the following steps:

Let $D = \{L, U\}$. D is the entire training dataset. $L = \{x_i, c_i\}$ is the labelled dataset with $i = 1, 2, \dots, M$, where M is the size of the labelled dataset and c is the class label of sample x . While $U = \{x_j, 0\}$ is the unlabelled dataset with $j = 1, 2, \dots, N$, where N is the size of unlabelled dataset and 0 denotes an unlabelled point.

- (1) Given L and $L' = \{\emptyset\}$ where L' represents an empty set of newly labelled data.
- (2) Present L to the GNG algorithm and train the network with only with L .
- (3) Label all the nodes of GNG network according to L .
- (4) Present x_j from U iteratively and compute the Euclidean distance between x_j and every nodes of GNG network:

$$\text{Distance} = \|w_n - x_i\|^2$$

- (5) Label x_j according to the class label of the winning node. Remove x_j from the current unlabelled dataset, U and add x_j into the newly labelled dataset L' .
- (6) If all unlabelled data has been labelled, go to (7), otherwise go back to (4).
- (7) Check labels of L' , if they become stable during successive iterations, go to step (8). Otherwise go back to step (4).
- (8) Present L and L' together to the GNG classifier and evaluate new classification performance.

It is important to include at least one input from each class when L is trained with GNG classifier. No representative from each class will result in biased labels to the newly labelled inputs that will reduce the overall performance of the classifier trained on L and L' . GNG parameters in this proposed algorithm are chosen to be constant for both independent GNG classifiers to ensure a stable and unbiased result.

4 Experimental Results

4.1 The Datasets

An artificially-generated dataset was used first to illustrate the details of the proposed SSGNG algorithm. The algorithm was then trained with the ORL face database to evaluate its accuracy and potential in face recognition and classification application.

Artificial Dataset

The artificial dataset consists of three Gaussian distributions, each representing a class. The dataset consists of 300 points with 100 points for each class. The distributions are centred at $[0, 5]$, $[-1.5, 1]$ and $[2, -1.5]$, with respective covariance matrix $[1 \ -0.0291; -0.0291 \ 4]$, $[1 \ -0.0061; -0.0061 \ 3]$ and $[1 \ -0.2251; -0.2251 \ 3]$. As can be seen in Fig. 1(a), samples from three classes are slightly overlapped and the class boundaries are not really visible. This is to analyse the proposed algorithm in terms of overlap and complexity of class boundary, as there is often a strong overlap between classes in practical datasets such as the ORL face data. Thus this dataset was also generated to simulate the complexity of ORL face database input distributions.

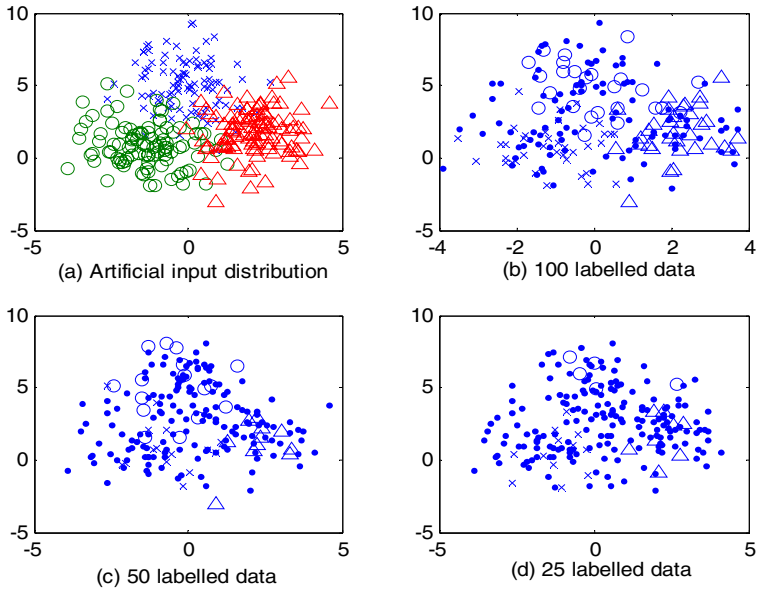


Fig. 1. (a) Scatter plot of all training data, D . (b), (c) and (d) show scatter plots of labelled and unlabelled data, $L+U$ with 100, 50 and 25 labelled samples respectively. The labelled data are marked with triangles, circles and crosses while unlabelled data are marked with dots.

The artificial dataset was randomly partitioned into the training set, D of 200 data points and the testing set, T of 100 data points. The training set was then randomly partitioned again into the labelled (L) and the unlabelled (U) datasets. The labels of the unlabelled dataset were permanently removed and were never used again in the algorithm.

The labelled and unlabelled data is split into ratios to investigate the effect of unlabelled data as well as the labelled data towards the classification accuracy. The ratio of unlabelled data is varied to evaluate the performance of the classifier with different amount of unlabelled data in the training set. For this experiment, the amounts of unlabelled data are chosen to be either 100, 50 or 25. Fig. 1(b)-(d) show the scatter plots of the labelled and unlabelled data of these three cases. Labelled points are marked with triangles, circles and crosses representing three classes; while unlabelled points are marked with small dots.

ORL Face Database

The ORL face database consists of 40 subjects, 10 different frontal images for each subject [9]. All images were taken against a dark homogenous background with upright, full-frontal position. The images are varied either on lightning, facial expressions or facial details. The size of each image is 92×112 pixels but for the sake of shorter training time, all images were resized to 46×56 pixels.



Fig. 2. Examples of ORL face database. These images portray the differences and similarities in facial details, facial expressions and lightning of the face images.

In this experiment, the ORL dataset are randomly partitioned to have 6 training images and 4 testing images for each subject. The training dataset are then randomly partitioned to have a varied number of either 3, 2 or 1 labelled data for each subject to see the effect of different amount of both unlabelled and labelled data for the proposed SSGNG algorithm. Fig. 2 shows 6 of 40 subjects of the ORL face database with variation of facial details and expressions and lightning. It can also be seen that there are similarities between subjects that contributes to the class overlapping.

4.2 Results and Observations

The experiment was setup to observe if the use of unlabelled data in combination with labelled data could improve the GNG classification accuracy for face recognition. The artificial dataset was trained and tested to verify the proposed algorithm before it was implemented and tested in the face recognition and classification system. As what have been described in the previous section, the artificial dataset and the ORL face database were being tested with varied numbers of labelled samples. The classification accuracy was measured by how many of data points or images from the testing set were correctly assigned to its class. To determine which class that would be assigned to the presented testing sample, the Euclidean distance between the data and each weight in the network was measured. The node with the weight having the shortest distance would be the winning node and the testing sample was assigned to the class of that particular node. The assigned classes of the testing dataset were then compared to their original classes to determine the classification accuracy. The parameters for the GNG algorithm were initialised to produce the best classification

Table 1. Classification rate on the test set for the artificial dataset. L only denotes training with labelled data only, while $L+L'$ denotes training with labelled and unlabelled data.

Training dataset size		Classification accuracy	
L	U	L only	$L + L'$
100	100	81.1%	82.7%
50	150	81%	82.6%
25	175	76.8%	80.45%

Table 2. Classification rate on the test set for the face recognition (ORL database). L only denotes training with labelled data only, while $L+L'$ denotes training with labelled and unlabelled data.

Training dataset size		Classification accuracy	
L	U	L only	$L + L'$
3	3	82.88%	83.32%
2	4	72.50%	75.19%
1	5	60.88%	63.63%

performance. Same parameters were used throughout this experiment to ensure an unbiased result. The setting of the parameters was chosen on trial-and-error basis. The details of the parameters are listed in Section 2. The growth of the GNG network is stopped when the network size reaches 100 nodes for the artificial dataset and 120 nodes for ORL Face database for simplicity, though more complicated validation process can be used.

Table 1 shows the percentage of classification accuracy for the test dataset of artificial data and Table 2 shows the classification accuracy for the face recognition. The results are the average percentage of the classification performance over 10 independent experiments. L and U in both tables denote the sizes of labelled data and unlabelled data, respectively.

From the results, it can be seen that the use of unlabelled data in the proposed SSSNG algorithm markedly improve the performance of the classifier. Greater improvements were made on fewer labelled samples, as shown in Tables 1 and 2. The classification performance reduces with fewer labelled data when the classifier is trained on labelled data only. However, the classification accuracy is significantly boosted with more unlabelled data. For the case with only 1 labelled face image for each subject, there is an average of 3-4% improvement to the classifier when compared to an average of 1-2% improvement with more labelled data. The results on the artificial data show similar performance improvements. The largest performance enhancement is achieved with the least available labelled data. In summary, unlabelled data can undoubtedly help improve the classification performance; and the more unlabelled data the greater improvement of the classification.

5 Conclusions

In this paper, a semi-supervised growing neural gas (SSGNG) is proposed for training classifiers with both labelled and unlabelled datasets or a partially labelled dataset. The classifier is first trained on the labelled data and then gradually unlabelled data is classified and added to the training data. The proposed SSGNG algorithm is demonstrated, on both artificial and real datasets, to significantly boost the classification rate with the use of unlabelled data. The improvement is particularly great when the labelled dataset is small or the unlabelled dataset is large. The SSGNG algorithm is computationally efficient and easy to implement. Further work will compare and incorporate with supervised learning algorithms such as support vector machines.

References

1. Fritzke, B.: A growing neural gas network learns topology. In: *Advances in Neural Information Processing Systems*, vol. 7. MIT Press, Cambridge (1995)
2. Kohonen, T.: *Self-Organizing Maps*. Springer, Heidelberg (1990)
3. Martinez, T.M., Berkovich, S.G., Schulten, K.J.: Neural gas – network for vector quantization and its application to time-series prediction. *IEEE Transactions on Neural Networks* 4, 558–569 (1993)
4. Fritzke, B.: Growing cell structures – a self-organising network for unsupervised and supervised Learning. *Neural Networks* 7(1994), 144–1460 (1994)
5. Martinez, T.: Competitive Hebbian learning rule forms perfectly topology preserving maps. In: *Proc. International Conference on Artificial Neural Networks (ICANN 1993)*, pp. 426–438 (1993)
6. Bouchachia, A.: Learning with partly labeled data. *Neural Computing and Application* 16, 267–293 (2006)
7. Dara, R., Kremer, S.C., Stacey, D.A.: Clustering unlabelled data with SOMs improves classification of labelled real-world data. In: *Proc. IEEE World Congress on Computational Intelligence*, pp. 2237–2242 (2002)
8. Seeger, M.: Learning with labelled and unlabelled data, Technical Report, Edinburgh University (2001)
9. Samaria, F., Harter, A.: Parameterisation of a stochastic model for human face identification (ORL face database). In: *2nd IEEE Workshop on Applications of Computer Vision*, Olivetti Research Laboratory (1994)

Author Index

- Ahmad, Khurshid 464
Ahmed, Chowdhury Farhan 193, 258
Amin, Md. Faijul 40
Araujo, Marcos Paulo Mello 32

Ban, Sang-Woo 88, 96
Barman, Paresh Chandra 120
Baruque, Bruno 491
Bedingfield, Susan 210
Borland, Ron 210
Bosin, Andrea 306
Bustillo, Andrés 498

Cai, Shengzhen 338
Carrasco-Ochoa, J. Ariel 282
Carrero, Francisco 346
Chang, Hsin-Yun 112
Chen, Hao 72
Chen, Jing 387
Chen, Qiaona 9
Chen, Xing 201
Cheung, Yiu-ming 473
Cho, Sung-Bae 156, 225
Choi, Inae 172
Choi, Kee-Hyun 412
Choi, Key-Sun 241
Choi, Kyuwan 330
Choi, Seungjin 140
Chung, Yong-Joo 49
Cichocki, Andrzej 330
Coghill, Ken 210
Corchado, Emilio 491, 498, 508
Cortizo, José Carlos 346
Costa, José A.F. 483
Cruz-Barbosa, Raúl 266
Curiel, Leticia 498

D.V.L.N., Somayajulu 250
Dashevskiy, Mikhail 274
Dehuri, Satchidananda 156
Deng, Yafeng 72
Dessi, Nicoletta 306
Ding, Xiaojiang 210
Dong, Yutao 387
Du, Liang 473
Dyson, Matthew 370

Franco-Arcega, Anilu 282
Fugini, Mariagrazia 306
Fyfe, Colin 452, 459

Gan, John Q. 370
Gómez, José María 346
Gonçalves, Márcio L. 483
González, Javier 491

Ha, IlKyu 436
Hamam, Y. 290
Han, Chang-Wook 80, 217
He, Xueming 128
Herrero, Álvaro 508
Herrmann, J. Michael 354, 362
Hu, Yujin 128
Hu, Zhaoguang 404, 420
Huang, Chi-Chun 112
Huang, Jin-Xia 241
Hwang, Byungku 96
Hwang, Keum-Sung 225

Ihrke, Matthias 354
Iqbal, Nadeem 104
Islam, Md. Monirul 40

Jang, Eunsong 452
Jeong, Byeong-Soo 193, 258
Jiao, Xiaoyou 404
Jiménez, Alfredo 508
Jimoh, A.A. 65
Jin, Feng 24
Jordaan, Jaco 65, 290

Kadampur, Mohammad Ali 250
Kamimura, Ryotaro 148
Kang, Byunguk 436
Kearney, Colm 464
Kim, Bumhwi 88
Kim, Hokun 459
Ko, Hanseok 452, 459
Kobayashi, Shigenobu 1
Kriksciuniene, Dalia 444
Kurien, A.M. 290

- Lee, Geon-Ha 412
 Lee, Minhø 88, 96
 Lee, Seung-Hyun 412
 Lee, Soo-Young 104, 120
 Lee, Young-Koo 193, 258
 Li, Chenggang 128
 Li, Chuan 387
 Li, Jian 57
 Li, Shenghong 57
 Liao, Xinfei 188
 Lin, Pan 338
 Liu, Mingrong 201
 Liu, Yicen 201
 Lu, Lei 164
 Luo, Zhiyuan 274

 Martínez-Trinidad, J. Fco 282
 Mishra, Bijan Bihari 156
 Mittal, Gauri S. 128
 Miyazaki, Kazuteru 1
 Mohd Zaki, Shireen 525
 Mourelle, Luiza de Macedo 32
 Murase, Kazuyuki 40

 Nedjah, Nadia 32
 Netto, Márcio L.A. 483

 Park, Sun 298
 Pes, Barbara 306
 Petrovic-Lazarevic, Sonja 210

 Roberts, Stephen J. 370
 Rovira, Jordi 491
 Ryu, Pum-Mo 241

 Sakalauskas, Virgilijus 444
 Sakr, Sherif 378
 Sánchez-Díaz, Guillermo 282
 Sattar, Md. Abdus 40
 Schrobsdorff, Hecke 354
 Sedano, Javier 498
 Shan, Baoguo 420
 Shim, JeongYon 314
 Shin, Dong-Ryeol 412
 Shin, Hyunjung 172
 Shiu, King Loong 428
 Siti, M.W. 65
 Su, Bo 57

 Sun, Shiliang 9, 24, 321
 Szeto, K.Y. 428

 Taşdemir, Kadim 180
 Tan, Hauchun 72
 Tan, Xiandong 420
 Tanbeer, Syed Khairuzzaman 193, 258
 Tao, Limin 188

 Van Wyk, B.J. 290
 Vellido, Alfredo 266
 Villar, José Ramón 498
 Voultzidou, Marotesa 362

 Wang, Shilin 57
 Wang, Shilong 387
 Whangbo, TaegKeun 233
 Wu, Kuo-Chuan 112
 Wu, Shengli 164

 Xiang, Liang 201
 Xu, Minjie 420
 Xu, Zeshui 17

 Yang, Cheng-Hong 112
 Yang, Qing 201
 Yang, Simon X. 128
 Yang, Zheng Rong 516
 Yao, Danhong 57
 Yeh, Chung-Hsing 210
 Yin, Hujun 525
 Yoo, Jiho 140
 Yoon, Ji Won 370
 Yoon, Yoe-Jin 412
 You, Xinge 473
 Young, David 210
 Yu, Gang 338
 Yun, Chang-Joo 210

 Zeng, Xiaoqin 164
 Zhang, Du 395
 Zhang, Jian Ying 210
 Zhang, Qian 233
 Zhang, Rong 128
 Zhang, Xianming 387
 Zheng, Chaoxin 464
 Zhong, Shuiming 164
 Zhu, Hong 395