

Tran Khanh Dang · Roland Wagner
Josef Küng · Nam Thoai
Makoto Takizawa · Erich Neuhold (Eds.)

LNCS 9446

Future Data and Security Engineering

Second International Conference, FDSE 2015
Ho Chi Minh City, Vietnam, November 23–25, 2015
Proceedings

FDSE 2015



Springer

Commenced Publication in 1973

Founding and Former Series Editors:

Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Editorial Board

David Hutchison

Lancaster University, Lancaster, UK

Takeo Kanade

Carnegie Mellon University, Pittsburgh, PA, USA

Josef Kittler

University of Surrey, Guildford, UK

Jon M. Kleinberg

Cornell University, Ithaca, NY, USA

Friedemann Mattern

ETH Zurich, Zürich, Switzerland

John C. Mitchell

Stanford University, Stanford, CA, USA

Moni Naor

Weizmann Institute of Science, Rehovot, Israel

C. Pandu Rangan

Indian Institute of Technology, Madras, India

Bernhard Steffen

TU Dortmund University, Dortmund, Germany

Demetri Terzopoulos

University of California, Los Angeles, CA, USA

Doug Tygar

University of California, Berkeley, CA, USA

Gerhard Weikum

Max Planck Institute for Informatics, Saarbrücken, Germany

More information about this series at <http://www.springer.com/series/7409>

Tran Khanh Dang · Roland Wagner
Josef Küng · Nam Thoai
Makoto Takizawa · Erich Neuhold (Eds.)

Future Data and Security Engineering

Second International Conference, FDSE 2015
Ho Chi Minh City, Vietnam, November 23–25, 2015
Proceedings

Editors

Tran Khanh Dang
Ho Chi Minh City University of Technology
Ho Chi Minh City
Vietnam

Nam Thoai
Ho Chi Minh City University of Technology
Ho Chi Minh City
Vietnam

Roland Wagner
Johannes Kepler University Linz
Linz
Austria

Makoto Takizawa
Hosei University
Tokyo
Japan

Josef Küng
Johannes Kepler University Linz
Linz
Austria

Erich Neuhold
University of Vienna
Vienna
Austria

ISSN 0302-9743

ISSN 1611-3349 (electronic)

Lecture Notes in Computer Science

ISBN 978-3-319-26134-8

ISBN 978-3-319-26135-5 (eBook)

DOI 10.1007/978-3-319-26135-5

Library of Congress Control Number: 2015953241

LNCS Sublibrary: SL3 – Information Systems and Applications, incl. Internet/Web, and HCI

Springer Cham Heidelberg New York Dordrecht London

© Springer International Publishing Switzerland 2015, corrected publication 2017

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made.

Printed on acid-free paper

Springer International Publishing AG Switzerland is part of Springer Science+Business Media
(www.springer.com)

Preface

In this volume we present the accepted contributions for the Second International Conference on Future Data and Security Engineering (FDSE 2015). The conference took place during November 23–25, 2015, in Ho Chi Minh City, Vietnam, at HCMC University of Technology, among the most famous and prestigious universities in Vietnam.

The annual FDSE conference is a premier forum designed for researchers, scientists, and practitioners interested in state-of-the-art and state-of-the-practice activities in data, information, knowledge, and security engineering to explore cutting-edge ideas, present and exchange their research results and advanced data-intensive applications, as well as to discuss emerging issues on data, information, knowledge, and security engineering.

The call for papers resulted in the submission of 88 papers. A rigorous and peer-review process was applied to all of them. This resulted in 20 full (including keynote speeches) and three short accepted papers (acceptance rate: 26.1 %), which were presented at the conference. Every paper was reviewed by at least three members of the international Program Committee, who were carefully chosen based on their knowledge and competence. This careful process resulted in the high quality of the contributions published in this volume. The accepted papers were grouped into the following sessions:

- Big data analytics and massive dataset mining
- Security and privacy engineering
- Crowdsourcing and social network data analytics
- Sensor databases and applications in smart home and city
- Emerging data management systems and applications
- Context-based data analysis and applications
- Data models and advances in query processing

In addition to the papers selected by the Program Committee, three internationally recognized scholars delivered keynote speeches: “An Empirical Study of the Attack Potential of Vulnerabilities,” presented by Professor Fabio Massacci from University of Trento, Italy; “The Asymmetric Architecture: A Privacy by Design Distributed Computing Architecture,” presented by Professor Benjamin Nguyen from INSA Centre Val de Loire, France; and “Modelling Sensible Business Processes,” presented by Associate Professor Pedro Antunes from Victoria University of Wellington, New Zealand.

In the first keynote speech, Professor Massacci talked about the attack potential of vulnerabilities and introduced a new estimator used as an aid for vulnerability prioritization. The abstract of the speech is briefly summarized as follows: “Vulnerability exploitation is reportedly one of the main attack vectors against computer systems. Characterization and assessment of vulnerabilities is therefore central to any IT security management activity. In particular, identifying *ex-ante* which vulnerabilities are most likely to be exploited (i.e., represent higher risk) is an open issue. In this talk, we identify trends in the volume of attacks in terms of the impact of vulnerability and complexity. As a result, we derive two possible ‘organizing principles’ for vulnerability

assessment and characterization that may prove useful to be integrated in current security management protocols and best practices. Over this notion we introduce an ‘attack potential’ estimator that reliably estimates the potential volume of attacks the vulnerability may receive in the wild. Our estimator can be used as an aid for vulnerability prioritization when deciding which vulnerability to fix first”.

In the second keynote speech, Professor Nguyen discussed important issues relevant to privacy in distributed computing architecture. The main contents of the speech are summarized as follows: “Today, there is a wide consensus that individuals should have increased control on how their personal data are collected, managed, and shared. Yet there is no appropriate technical solution to implement such personal data services: centralized solutions sacrifice security for innovative applications, while decentralized solutions sacrifice innovative applications for security. In previous works, we argued that the advent of secure hardware in all personal IT devices, on the edges of the Internet, could trigger a sea change, called the Trusted Cells paradigm: personal data servers running on secure smart phones, set-top boxes, secure portable tokens or smart cards to form a global, decentralized data platform that both provides security and encourages innovative applications. In this talk, we describe how to run distributed computing on an infrastructure composed of a vast set of low-powered, highly secure Trusted Cells, and an untrusted Supporting Server Infrastructure. We call this infrastructure the Asymmetric Architecture. The results include computing SQL Group By queries, anonymization algorithms, or even Map/Reduce on this architecture”.

In the last talk Associate Professor Antunes presented a brand-new concept of sensible business process, which balances the level of control between machines and humans within the BPM systems. The abstract of this work is as follows: “In this talk, we develop the concept of sensible business process, which appears in opposition to the more traditional concept of mechanistic business process that is currently supported by most business process modelling languages and tools. A sensible business process is founded on a rich model and affords predominant human control. Having previously developed a modelling tool supporting this concept, in this talk we report on a set of experiments with the tool. The results obtained show that the approach (1) captures richer information about business processes; (2) contributes to knowledge sharing in organizations; and (3) generates better process models.”

The success of FDSE 2015 was the result of the efforts of many people, to whom we would like to express our gratitude. First, we would like to thank all authors who submitted papers to FDSE 2015, especially the invited speakers. We would also like to thank the members of the committees and external reviewers for their timely reviewing and lively participation in the subsequent discussion in order to select such high-quality papers published in this volume. Last but not least, we thank the Faculty of Computer Science and Engineering, HCMC University of Technology, for hosting FDSE 2015.

November 2015

Tran Khanh Dang
 Roland Wagner
 Josef Küng
 Nam Thoai
 Makoto Takizawa
 Erich Neuhold

Organization

General Chair

Roland Wagner

Johannes Kepler University Linz, Austria

Steering Committee

Elisa Bertino

Purdue University, USA

Kazuhiko Hamamoto

Tokai University, Japan

Abdelkader Hameurlain

Paul Sabatier University, Toulouse, France

M-Tahar Kechadi

University College Dublin, Ireland

Dieter Kranzlmüller

Ludwig Maximilians University, Germany

Josef Küng

Johannes Kepler University Linz, Austria

Atsuko Miyaji

Japan Advanced Institute of Science and Technology,
Japan

Beng Chin Ooi

National University of Singapore, Singapore

Nam Thoi

HCMC University of Technology, Vietnam

A. Min Tjoa

Technical University of Vienna, Austria

Shigeki Yamada

National Institute of Informatics, Japan

Program Committee Chairs

Tran Khanh Dang

HCMC University of Technology, Vietnam

Makoto Takizawa

Hosei University, Japan

Erich Neuhold

University of Vienna, Austria

Publicity Chairs

Phan Trong Nhan

Johannes Kepler University Linz, Austria

Tran Minh Quang

National Institute of Informatics, Japan,
and HCMC University of Technology, Vietnam

Quan Thanh Tho

HCMC University of Technology, Vietnam

Hoang Tam Vo

SAP Research and Innovation, Singapore

Local Organizing Committee

Tran Khanh Dang

HCMC University of Technology, Vietnam (Chair)

Nam Thoi

HCMC University of Technology, Vietnam (Co-chair)

Hong Thanh Luan

Can Tho University of Technology, Vietnam

Tran Tri Dang

HCMC University of Technology, Vietnam

Tran Ngoc Thinh

HCMC University of Technology, Vietnam

Truong Quynh Chi	HCMC University of Technology, Vietnam
Tran Thi Que Nguyet	HCMC University of Technology, Vietnam
Nguyen Thi Ai Thao	HCMC University of Technology, Vietnam
Le Thi Kim Tuyen	Sungkyunkwan University, South Korea, and HCMC University of Technology, Vietnam
Ngo Chan Nam	Data Security Applied Research Lab, HCMUT, Vietnam
Nguyen Thanh Tung	HCMC University of Technology, Vietnam
Van Duc Son Ha	HCMC University of Technology, Vietnam
La Hue Anh	HCMC University of Technology, Vietnam

Program Committee

Pedro Antunes	Victoria University of Wellington, New Zealand
Stephane Bressan	National University of Singapore, Singapore
Hyunseung Choo	Sungkyunkwan University, South Korea
Somsak Choomchuay	King Mongkut's Institute of Technology Ladkrabang, Thailand
Agostino Cortesi	Università Ca' Foscari Venezia, Italy
Nguyen Tuan Dang	University of Information Technology, VNUHCM, Vietnam
Tran Cao De	Can Tho University, Vietnam
Thanh-Nghi Do	Can Tho University, Vietnam
Nguyen Van Doan	Japan Advanced Institute of Science and Technology, Japan
Dirk Draheim	University of Innsbruck, Austria
Verena Geist	Software Competence Center Hagenberg, Austria
Raju Halder	Indian Institute of Technology Patna, India
Tran Van Hoai	HCMC University of Technology, Vietnam
Nguyen Viet Hung	University of Trento, Italy
Nguyen Quoc Viet Hung	École polytechnique fédérale de Lausanne, Switzerland
Tran Nguyen Hoang Huy	University of Vienna, Austria
Trung-Hieu Huynh	Industrial University of Ho Chi Minh City, Vietnam
Van-Nam Huynh	Japan Advanced Institute of Science and Technology, Japan
Ryutaro Ichise	National Institute of Informatics, Japan
Tomohiko Igasaki	Kumamoto University, Japan
Koichiro Ishibashi	University of Electro-Communications, Japan
Hiroshi Ishii	Tokai University, Japan
Kazuhiko Hamamoto	Tokai University, Japan
Abdelkader Hameurlain	Paul Sabatier University, Toulouse, France
Eiji Kamioka	Shibaura Institute of Technology, Japan
M-Tahar Kechadi	University College Dublin, Ireland
Le Duy Khanh	Data Storage Institute, Singapore
Surin Kittitornkun	King Mongkut's Institute of Technology Ladkrabang, Thailand

Andrea Ko	Corvinus University of Budapest, Hungary
Hilda Kosorus	Johannes Kepler University Linz, Austria
Lam Son Le	HCMC University of Technology, Vietnam
Tan Kian Lee	National University of Singapore, Singapore
Fabio Massacci	University of Trento, Italy
Hoang Duc Minh	National Physical Laboratory, UK
Atsuko Miyaji	Japan Advanced Institute of Science and Technology, Japan
Takumi Miyoshi	Shibaura Institute of Technology, Japan
Hiroaki Morino	Shibaura Institute of Technology, Japan
Thanh Binh Nguyen	HCMC University of Technology, Vietnam
Benjamin Nguyen	Institut National des Sciences Appliquées Centre Val de Loire, France
Khoa Nguyen	National ICT Australia, Australia
Vu Thanh Nguyen	University of Information Technology, VNUHCM, Vietnam
Phan Trong Nhan	Johannes Kepler University Linz, Austria
Eric Pardede	La Trobe University, Australia
Cong Duc Pham	University of Pau, France
Nguyen Khang Pham	Can Tho University, Vietnam
Phung Huu Phu	University of Gothenburg, Sweden, and University of Illinois at Chicago, USA
Tran Minh Quang	National Institute of Informatics, Japan, and HCMC University of Technology, Vietnam
Le Thanh Sach	HCMC University of Technology, Vietnam
Akbar Saiful	Institute of Technology Bandung, Indonesia
Tran Le Minh Sang	WorldQuant LLC, USA
Christin Seifert	University of Passau, Germany
Erik Sonnleitner	Johannes Kepler University Linz, Austria
Reinhard Stumptner	Software Competence Center Hagenberg, Austria
Tran Ngoc Thinh	HCMC University of Technology, Vietnam
Quoc Cuong To	Inria Rocquencourt, Versailles, France
Shigenori Tomiyama	Tokai University, Japan
Ha-Manh Tran	International University, Vietnam
Tuan Anh Truong	University of Trento, Italy
Hong Linh Truong	Vienna University of Technology, Austria
Tran Minh Triet	HCMC University of Natural Sciences, Vietnam
Truong Minh Nhat Quang	Can Tho University of Technology, Vietnam
Osamu Uchida	Tokai University, Japan
Hoang Tam Vo	SAP Research and Innovation, Singapore
Pham Tran Vu	HCMC University of Technology, Vietnam
Edgar Weippl	Technical University of Vienna, Austria

External Reviewers

Nguyen Ngoc Thien An	University College Dublin, Ireland
Vo Thi Ngoc Chau	HCMC University of Technology, Vietnam
Truong Quang Hai	HCMC University of Technology, Vietnam
Thuan Nguyen Hoang	Victoria University of Wellington, New Zealand
Huynh Van Quoc Phuong	HCMC University of Technology, Vietnam
Le Thi Bao Thu	Inria Rocquencourt, Versailles, France
Le Thanh Van	HCMC University of Technology, Vietnam

Contents

Big Data Analytics and Massive Dataset Mining

- Random Local SVMs for Classifying Large Datasets 3
Thanh-Nghi Do and François Poulet
- An Efficient Document Indexing-Based Similarity Search in Large Datasets 16
*Trong Nhan Phan, Markus Jäger, Stefan Nadschläger, Josef Küng,
and Tran Khanh Dang*
- Using Local Rules in Random Forests of Decision Trees 32
Thanh-Nghi Do
- A Term Weighting Scheme Approach for Vietnamese Text Classification 46
*Vu Thanh Nguyen, Nguyen Tri Hai, Nguyen Hoang Nghia,
and Tuan Dinh Le*

Security and Privacy Engineering

- Fault Data Analytics Using Decision Tree for Fault Detection 57
Ha Manh Tran, Sinh Van Nguyen, Son Thanh Le, and Quy Tran Vu
- Evaluation of Reliability and Security of the Address Resolution Protocol 72
Elvia León, Brayan S. Reyes Daza, and Octavio J. Salcedo Parra

Crowdsourcing and Social Network Data Analytics

- Establishing a Decision Tool for Business Process Crowdsourcing 85
*Nguyen Hoang Thuan, Pedro Antunes, David Johnstone,
and Nguyen Huynh Anh Duy*
- Finding Similar Artists from the Web of Data: A PageRank Based Semantic
Similarity Metric 98
Phuong T. Nguyen and Hong Anh Le
- Opinion Analysis in Social Networks Using Antonym Concepts on Graphs 109
Hiram Calvo

Sensor Databases and Applications in Smart Home and City

- Traffic Speed Data Investigation with Hierarchical Modeling 123
Tomonari Masada and Atsuhiko Takasu

An Effective Approach to Background Traffic Detection	135
<i>Quang Tran Minh</i>	
An Approach for Developing Intelligent Systems in Smart Home Environment	147
<i>Tran Nguyen Minh-Thai and Nguyen Thai-Nghe</i>	
Emerging Data Management Systems and Applications	
Modelling Sensible Business Processes	165
<i>David Simões, Nguyen Hoang Thuan, Lalitha Jonnavithula, and Pedro Antunes</i>	
Contractual Proximity of Business Services	183
<i>Lam-Son Lê</i>	
Energy-Efficient VM Scheduling in IaaS Clouds.	198
<i>Nguyen Quang-Hung and Nam Thoai</i>	
Multi-diagram Representation of Enterprise Architecture: Information Visualization Meets Enterprise Information Management	211
<i>Lam-Son Lê</i>	
Enhancing the Quality of Medical Image Database Based on Kernels in Bandelet Domain.	226
<i>Nguyen Thanh Binh</i>	
Information Systems Success: A Literature Review	242
<i>Thanh D. Nguyen, Tuan M. Nguyen, and Thi H. Cao</i>	
Context-Based Data Analysis and Applications	
Facilitating the Design/Evaluation Process of Web-Based Geographic Applications: A Case Study with WINDMash.	259
<i>The Nhan Luong, Christophe Marquesuzaà, Patrick Etcheverry, Thierry Nodenot, and Sébastien Laborie</i>	
A Context-Aware Recommendation Framework in E-Learning Environment.	272
<i>Phung Do, Hung Nguyen, Vu Thanh Nguyen, and Tran Nam Dung</i>	
Automatic Evaluation of the Computing Domain Ontology	285
<i>Chien D.C. Ta and Tuoi Phan Thi</i>	

Data Models and Advances in Query Processing

Comics Instance Search with Bag of Visual Words 299
Duc-Hoang Nguyen, Minh-Triet Tran, and Vinh-Tiep Nguyen

Defining Membership Functions in Fuzzy Object-Oriented Database Model . . . 314
Doan Van Thang and Dang Cong Quoc

Erratum to: Facilitating the Design/Evaluation Process
of Web-Based Geographic Applications: A Case Study with WINDMash. E1
*The Nhan Luong, Christophe Marquesuzaà, Patrick Etcheverry,
Thierry Nodenot, and Sébastien Laborie*

Author Index 323

Big Data Analytics and Massive Dataset Mining

Random Local SVMs for Classifying Large Datasets

Thanh-Nghi Do¹(✉) and François Poulet²

¹ College of Information Technology, Can Tho University, Cantho 92100, Vietnam
dtngghi@cit.ctu.edu.vn

² University of Rennes I - IRISA, Campus de Beaulieu, 35042 Rennes Cedex, France
francois.poulet@irisa.fr

Abstract. We propose a new parallel ensemble learning algorithm of random local support vector machines, called krSVM for the effectively non-linear classification of large datasets. The random local SVM in the krSVM learning strategy uses k means algorithm to partition the data into k clusters, followed which it constructs a non-linear SVM in each cluster to classify the data locally in the parallel way on multi-core computers. The krSVM algorithm is faster than the standard SVM in the non-linear classification of large datasets while maintaining the classification correctness. The numerical test results on 4 datasets from UCI repository and 3 benchmarks of handwritten letters recognition showed that our proposed algorithm is efficient compared to the standard SVM.

Keywords: Support vector machines · Random local support vector machines · Large-scale non-linear classification

1 Introduction

In recent years, the SVM algorithm proposed by [1] and kernel-based methods have shown practical relevance for classification, regression and novelty detection. Successful applications of SVMs have been reported for various fields like face identification, text categorization and bioinformatics [2]. They become increasingly popular data analysis tools. In spite of the prominent properties of SVM, they are not favorable to deal with the challenge of large datasets. SVM solutions are obtained from quadratic programming (QP), so that the computational cost of a SVM approach is at least square of the number of training datapoints and the memory requirement making SVM impractical. There is a need to scale up learning algorithms to handle massive datasets on personal computers (PCs).

Our investigation is to propose a new parallel ensemble learning algorithm of random local SVM, called krSVM for the effectively non-linear classification of large datasets. Instead of building a global SVM model, as done by the classical algorithm which is very difficult to deal with large datasets, the krSVM algorithm constructs an ensemble of local ones that are easily trained by the standard

SVM algorithms. The random local SVM in the krSVM algorithm performs the training task with two steps. The first one is to use k means algorithm [3] to partition the data into k clusters, and then the second one is to learn a non-linear SVM in each cluster to classify the data locally in the parallel way on multi-core computers. The numerical test results on 4 datasets from UCI repository [4] and 3 benchmarks of handwritten letters recognition [5], MNIST [6, 7] show that our proposal is efficient compared to the standard SVM in terms of training time and accuracy. The krSVM algorithm is faster than the standard SVM in the non-linear classification of large datasets while maintaining the high classification accuracy.

The paper is organized as follows. Section 2 briefly introduces the SVM algorithm. Section 3 presents our proposed parallel algorithm of random local SVM for the non-linear classification of large datasets. Section 4 shows the experimental results. Section 5 discusses about related works. We then conclude in Sect. 6.

2 Support Vector Machines

Let us consider a linear binary classification task, as depicted in Fig. 1, with m datapoints x_i ($i = 1, \dots, m$) in the n -dimensional input space R^n , having corresponding labels $y_i = \pm 1$. For this problem, the SVM algorithms [1] try to find the best separating plane (denoted by the normal vector $w \in R^n$ and the scalar $b \in R$), i.e. furthest from both class +1 and class -1. It can simply

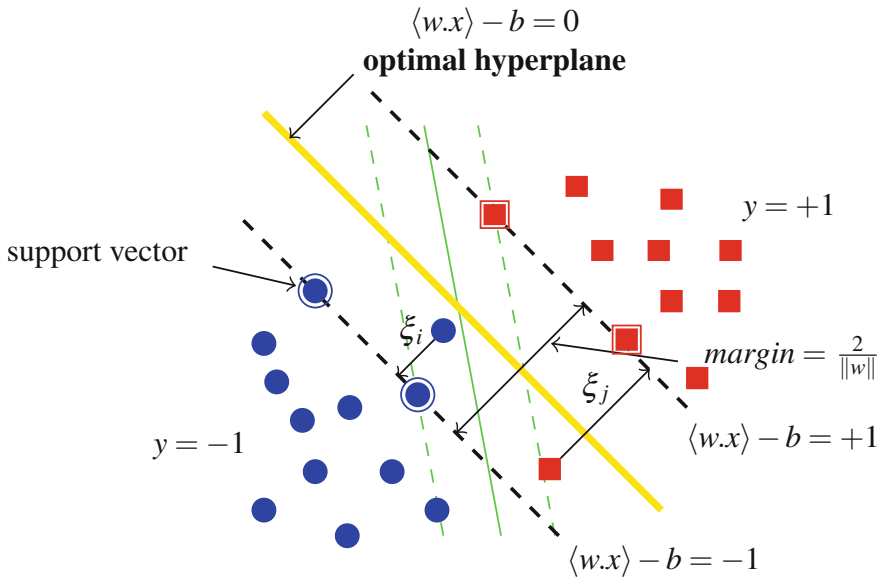


Fig. 1. Linear separation of the datapoints into two classes

maximize the distance or the margin between the supporting planes for each class ($x \cdot w - b = +1$ for class $+1$, $x \cdot w - b = -1$ for class -1). The margin between these supporting planes is $2/\|w\|$ (where $\|w\|$ is the 2-norm of the vector w). Any point x_i falling on the wrong side of its supporting plane is considered to be an error, denoted by z_i ($z_i \geq 0$). Therefore, SVM has to simultaneously maximize the margin and minimize the error. The standard SVMs pursue these goals with the quadratic programming of (1).

$$\begin{aligned} \min_{\alpha} (1/2) \sum_{i=1}^m \sum_{j=1}^m y_i y_j \alpha_i \alpha_j K \langle x_i, x_j \rangle - \sum_{i=1}^m \alpha_i \\ \text{s.t.} \begin{cases} \sum_{i=1}^m y_i \alpha_i = 0 \\ 0 \leq \alpha_i \leq C \quad \forall i = 1, 2, \dots, m \end{cases} \end{aligned} \quad (1)$$

where C is a positive constant used to tune the margin and the error and a linear kernel function $K \langle x_i, x_j \rangle = \langle x_i \cdot x_j \rangle$.

The support vectors (for which $\alpha_i > 0$) are given by the solution of the quadratic program (1), and then, the separating surface and the scalar b are determined by the support vectors. The classification of a new data point x based on the SVM model is as follows:

$$\text{predict}(x, SVMmodel) = \text{sign} \left(\sum_{i=1}^{\#SV} y_i \alpha_i K \langle x, x_i \rangle - b \right) \quad (2)$$

Variations on SVM algorithms use different classification functions [8]. No algorithmic changes are required from the usual kernel function $K \langle x_i, x_j \rangle$ as a linear inner product, $K \langle x_i, x_j \rangle = \langle x_i \cdot x_j \rangle$ other than the modification of the kernel function evaluation. We can get different support vector classification models. There are two other popular non-linear kernel functions as follows:

- a polynomial function of degree d :
 $K \langle x_i, x_j \rangle = (\langle x_i \cdot x_j \rangle + 1)^d$
- a RBF (Radial Basis Function): $K \langle x_i, x_j \rangle = e^{-\gamma \|x_i - x_j\|^2}$

SVMs are accurate models with practical relevance for classification, regression and novelty detection. Successful applications of SVMs have been reported for such varied fields including facial recognition, text categorization and bioinformatics [2].

3 Parallel Ensemble Learning Algorithm of Random Local Support Vector Machines

The study in [9] illustrated that the computational cost requirements of the SVM solutions in (1) are at least $O(m^2)$ (where m is the number of training

datapoints), making standard SVM intractable for large datasets. Learning a global SVM model on the full massive dataset is challenge due to the very high computational cost and the very large memory requirement.

Learning local SVM models. Our proposal in [10] deals with large datasets via the training of local SVM (denoted by kSVM). The main idea is to partition the full dataset into k clusters and then it is easily to learn a non-linear SVM in each cluster to classify the data locally. Figure 2 shows the comparison between a global SVM model (left part) and 3 local SVM models (right part), using a non-linear RBF kernel function with $\gamma = 10$ and a positive constant $C = 10^6$. In practice, k -means algorithm [3] is the most widely used partitional clustering algorithm because it is simple, easily understandable, and reasonably scalable [11]. Therefore, we propose to use k means algorithm to partition the full dataset into k clusters and the standard SVM (e.g. LibSVM [12]) to learn k local SVMs.

Let now examine the complexity of building k local SVM models with the kSVM algorithm. The full dataset with m individuals is partitioned into k balanced clusters (the cluster size is about $\frac{m}{k}$). Therefore, the complexity of k local SVM models is $O(k(\frac{m}{k})^2) = O(\frac{m^2}{k})$. This complexity analysis illustrates that learning k local SVM models in the kSVM algorithm¹ is faster than building a global SVM model (the complexity is at least $O(m^2)$).

It must be remarked that the parameter k is used in the kSVM to give a trade-off between the generalization capacity and the computational cost. In [13–15], Vapnik points out the trade-off between the capacity of the local learning system and the number of available individuals. In the context of k local SVM models, this point can be understood as follows:

- If k is large then the kSVM algorithm reduces significant training time (the complexity of kSVM is $O(\frac{m^2}{k})$). And then, the size of a cluster is small; The locality is extremely with a very low capacity.
- If k is small then the kSVM algorithm reduces insignificant training time. However, the size of a cluster is large; It improves the capacity.

It leads to set k so that the cluster size is large enough (e.g. 200 proposed by [14]).

Furthermore, the kSVM learns independently k local models from k clusters. This is a nice property for parallel learning. The parallel kSVM does take into account the benefits of high performance computing, e.g. multi-core computers or grids. The simplest development of the parallel kSVM algorithm is based on the shared memory multiprocessing programming model OpenMP [16] on multi-core computers. The parallel training of kSVM is described in Algorithm 1.

Prediction of a new individual using local SVM models. The k SVM – model = $\{(c_1, lsvm_1), (c_2, lsvm_2), \dots, (c_k, lsvm_k)\}$ is used to predict the class of

¹ It must be noted that the complexity of the kSVM approach does not include the k means clustering used to partition the full dataset. But this step requires insignificant time compared with the quadratic programming solution.

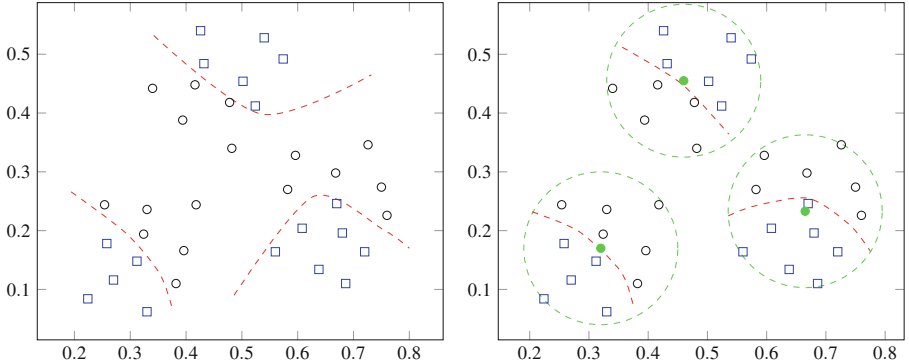


Fig. 2. Global SVM model (left part) versus local SVM models (right part)

a new individual x as follows. The first step is to find the closest cluster based on the distance between x and the cluster centers:

$$c_{NN} = \arg \min_c \text{distance}(x, c) \quad (3)$$

And then, the class of x is predicted by the local SVM model $lsvm_{NN}$ (corresponding to c_{NN}):

$$\text{predict}(x, kSVM\text{model}) = \text{predict}(x, lsvm_{NN}) \quad (4)$$

Ensemble of random local SVM models. The analysis of the trade-off between the generalization capacity and the computational cost illustrates that the local SVM models tries to speed up the training time of the global SVM model while reducing the generalization capacity. Due to this problem, we propose to construct the ensemble of random local SVM models to improve the generalization capacity of the local one. The main idea is based on the random forests proposed by Breiman [17]. The randomization is used for controlling high diversity between local SVM models². It leads to the improvement of the generalization capacity of the single one local SVM model. The ensemble of random local SVM (krSVM described in Algorithm 2) creates a collection of T random local SVMs (kSVM described in Algorithm 1) from bootstrap samples (sampling with replacement from the original dataset) using a randomly chosen subset of attributes. Thus, the complexity of krSVM is $O(T \frac{m^2}{k})$.

Furthermore, the krSVM constructs independently T random local SVM models (kSVM). It allows parallelizing the learning task with OpenMP [16] on multi-core computers.

The prediction class of a new individual x is the plurality class of the classification results obtained by T kSVM models.

² Two classifiers are diverse if they make different errors on new data points [18].

Algorithm 1. Local SVM algorithm (kSVM)

```

input :
    training dataset  $D$ 
    number of local models  $k$ 
    hyper-parameter of RBF kernel function  $\gamma$ 
     $C$  for tuning margin and errors of SVMs

output:
     $k$  local support vector machines models

1 begin
2   /* $k$ means performs the data clustering on  $D$ ;*/
3   creating  $k$  clusters denoted by  $D_1, D_2, \dots, D_k$  and
4   their corresponding centers  $c_1, c_2, \dots, c_k$ 
5   #pragma omp parallel for
6   for  $i \leftarrow 1$  to  $k$  do
7     /*learning local SVM model from  $D_i$ ;*/
8     |  $lsvm_i = svm(D_i, \gamma, C)$ 
9   end
10  return  $kSVM - model = \{(c_1, lsvm_1), (c_2, lsvm_2), \dots, (c_k, lsvm_k)\}$ 
11 end

```

4 Evaluation

We are interested in the performance of the new parallel ensemble learning algorithm of random local SVM (denoted by krSVM) for data classification. We have implemented krSVM in C/C++, OpenMP [16], using the highly efficient standard library SVM, LibSVM [12]. Our evaluation of the classification performance is reported in terms of correctness and training time. We are interested in the comparison obtained by our proposed krSVM with LibSVM.

All experiments are run on machine Linux Fedora 20, Intel(R) Core i7-4790 CPU, 3.6 GHz, 4 cores and 32 GB main memory.

Experiments are conducted with the 4 datasets collected from UCI repository [4] and the 3 benchmarks of handwritten letters recognition, including USPS [5], MNIST [6], a new benchmark for handwritten character recognition [7]. Table 1 presents the description of datasets. The evaluation protocols are illustrated in the last column of Table 1. Datasets are already divided in training set (Trn) and testing set (Tst). We used the training data to build the SVM models. Then, we classified the testing set using the resulting models.

We propose to use RBF kernel type in krSVM and SVM models because it is general and efficient [19]. We also tried to tune the hyper-parameter γ of RBF kernel (RBF kernel of two individuals x_i, x_j , $K[i, j] = \exp(-\gamma \|x_i - x_j\|^2)$) and the cost C (a trade-off between the margin size and the errors) to obtain a good accuracy. Furthermore, our krSVM uses 20 random local SVM models with the number of random attributes being one half full set. For the parameter k local models (number of clusters), we propose to set k so that each cluster has about 1000 individuals. The idea gives a trade-off between the generalization

Algorithm 2. Ensemble of random local SVM (krSVM)

```

input :
    training dataset  $D$ 
    number of kSVM models  $T$ 
     $rdims$  random attributes used in the kSVM model
     $k$  local models in the kSVM model
    hyper-parameter of RBF kernel function  $\gamma$ 
     $C$  for tuning margin and errors of SVMs

output:
     $T$  kSVM models

1 begin
2   #pragma omp parallel for
3   for  $t \leftarrow 1$  to  $T$  do
4     Sampling a bootstrap  $D_t$  (train set) from  $D$  using  $rdims$  random
       attributes)
5      $kSVM_t = kSVM(D_t, k, \gamma, C)$ 
6   end
7   return  $krSVM - model = \{kSVM_1, kSVM_2, \dots, kSVM_T\}$ 
8 end

```

Table 1. Description of datasets

ID	Dataset	Individuals	Attributes	Classes	Evaluation protocol
1	Opt. Rec. of Handwritten Digits	5620	64	10	3832 Trn - 1797 Tst
2	Letter	20000	16	26	13334 Trn - 6666 Tst
3	Isolet	7797	617	26	6238 Trn - 1559 Tst
4	USPS Handwritten Digit	9298	256	10	7291 Trn - 2007 Tst
5	A New Benchmark for Hand. Char. Rec	40133	3136	36	36000 Trn - 4133 Tst
6	MNIST	70000	784	10	60000 Trn - 10000 Tst
7	Forest Cover Types	581012	54	7	400000 Trn - 181012 Tst

capacity [15] and the computational cost. Table 2 presents the hyper-parameters of krSVM, kSVM and SVM in the classification.

The classification results of LibSVM, krSVM and kSVM on the 7 datasets are given in Table 3 and Figs. 3 and 4. As it was expected, our krSVM algorithm outperforms LibSVM in terms of training time. krSVM is about 1.5 times slower than kSVM. In terms of test correctness, our krSVM achieves very competitive performances compared to LibSVM. kSVM is less accurate than krSVM.

With 5 first small datasets, the improvement of krSVM is not significant. With large datasets, krSVM achieves a significant speed-up in learning. For MNIST dataset, krSVM is 18.61 times faster than LibSVM. Typically, Forest cover type dataset is well-known as a difficult dataset for non-linear SVM [20, 21]; LibSVM ran for 23 days without any result. krSVM performed this non-linear classification in 273.36 seconds with 97.07% accuracy.

Table 2. Hyper-parameters of krSVM, kSVM and SVM

ID	Datasets	γ	C	k
1	Opt. Rec. of Handwritten Digits	0.0001	100000	10
2	Letter	0.0001	100000	30
3	Isolet	0.0001	100000	10
4	USPS Handwritten Digit	0.0001	100000	10
5	A New Benchmark for Hand. Char. Rec	0.001	100000	50
6	MNIST	0.05	100000	100
7	Forest Cover Types	0.0001	100000	500

Table 3. Classification results in terms of accuracy (%) and training time (s)

ID	Datasets	Classification accuracy(%)			Training time(s)		
		LibSVM	krSVM	kSVM	LibSVM	krSVM	kSVM
1	Opt. Rec. of Handwritten Digits	98.33	97.61	97.05	0.58	0.54	0.21
2	Letter	97.40	97.16	96.14	2.87	1.94	0.5
3	Isolet	96.47	96.15	95.44	8.37	7.14	2.94
4	USPS Handwritten Digit	96.86	96.46	95.86	5.88	5.32	3.82
5	A New Benchmark for Hand. Char. Rec	95.14	94.77	92.98	107.07	91.72	35.7
6	MNIST	98.37	98.71	98.11	1531.06	82.26	45.5
7	Forest Cover Types	NA	97.07	97.06	NA	273.36	223.7

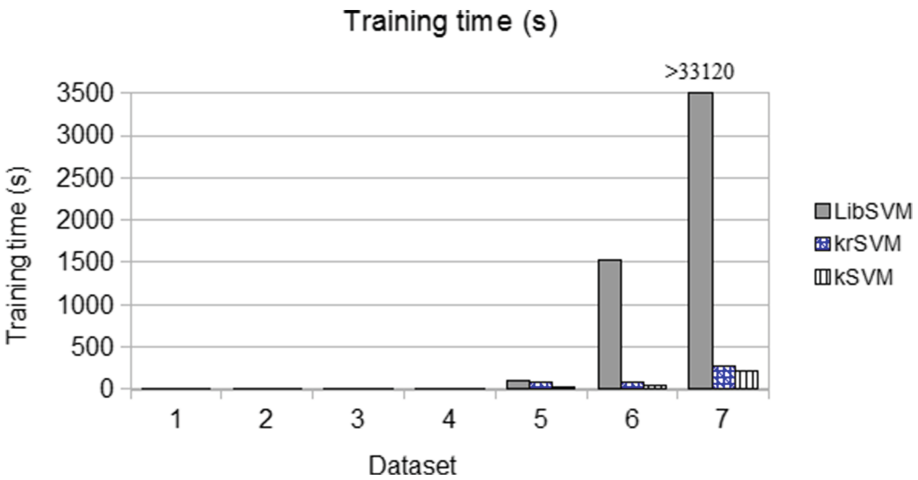


Fig. 3. Comparison of training time

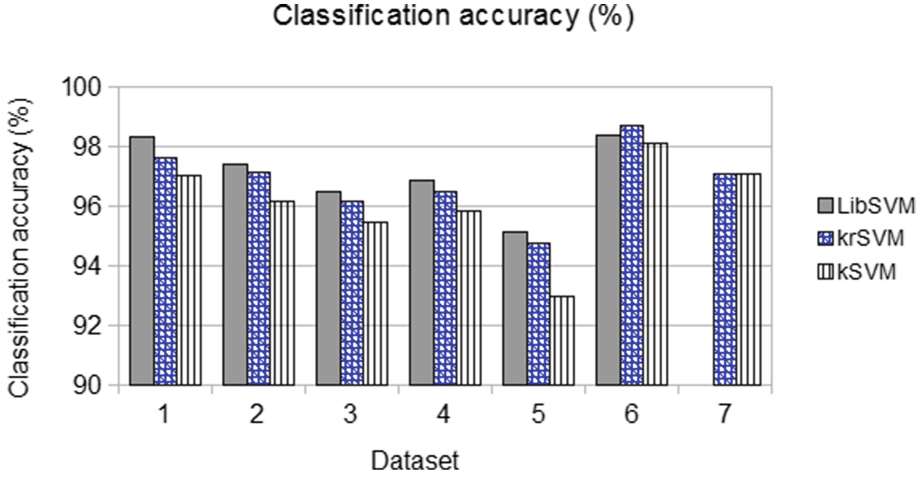


Fig. 4. Comparison of accuracy

5 Discussion on Related Works

Our proposal is in some aspects related to large-scale SVM learning algorithms. The improvements of SVM training on very large datasets include effective heuristic methods in the decomposition of the original quadratic programming into series of small problems [9, 12, 22, 23].

Mangasarian and his colleagues proposed to modify SVM problems to obtain new formulas, including Lagrangian SVM [24], proximal SVM [25], Newton SVM [26]. The Least Squares SVM proposed by Suykens and Vandewalle [27] changes standard SVM optimization to lead the new efficient SVM solver. And then, these algorithms only require solving a system of linear equations instead of a quadratic programming. This makes training time very short for linear classification tasks. More recent [28, 29] proposed the stochastic gradient descent methods for dealing with large scale linear SVM solvers. Their extensions proposed by [21, 30–33] aim at improving memory performance for massive datasets by incrementally updating solutions in a growing training set without needing to load the entire dataset into memory at once. The parallel and distributed algorithms [31, 33, 34] for the linear classification improve learning performance for large datasets by dividing the problem into sub-problems that execute on large numbers of networked PCs, grid computing, multi-core computers. Parallel SVMs proposed by [35] use GPU to speed-up training tasks. These algorithms are efficient for linear classification tasks.

Active SVM learning algorithms proposed by [20, 36–38] choose interesting datapoint subsets (active sets) to construct models, instead of using the whole dataset. SVM algorithms [21, 39–41] use the boosting strategy [42, 43] for the linear classification of very large datasets on standard PCs.

Our proposal of local SVM models aims at dealing with non-linear classification tasks. It is also related to local learning algorithms. The first paper of [44] proposed to use the expectation-maximization algorithm [45] for partitioning the training set into k clusters; for each cluster, a neural network is learnt to classify the individuals in the cluster. Local learning algorithms of Bottou & Vapnik [14] find k nearest neighbors of a test individual; train a neural network with only these k neighborhoods and apply the resulting network to the test individual. k -local hyperplane and convex distance nearest neighbor algorithms were also proposed in [46]. More recent local SVM algorithms include SVM-kNN [47], ALH [48], FaLK-SVM [49], LSVM [50], LL-SVM [51, 52], CSVN [53]. A theoretical analysis for such local algorithms discussed in [13] introduces the trade-off between the capacity of learning system and the number of available individuals. The size of the neighborhoods is used as an additional free parameters to control capacity against locality of local learning algorithms.

6 Conclusion and Future Works

We presented the new parallel ensemble learning algorithm of random local support vector machines that achieves high performances for the non-linear classification of large datasets. The training task of the random local SVM in the krSVM model is to partition the data into k clusters and then it constructs a non-linear SVM in each cluster to classify the data locally in the parallel way. The numerical test results on 4 datasets from UCI repository and 3 benchmarks of handwritten letters recognition showed that our proposed algorithm is efficient in terms of training time and accuracy compared to the standard SVM. An example of its effectiveness is given with the non-linear classification of Forest Cover Types dataset (having 400000 individuals, 54 attributes) into 7 classes in 273.36 seconds and 97.06% accuracy.

In the near future, we intend to provide more empirical test on large benchmarks and comparisons with other algorithms. A promising future research aims at improving the classification accuracy of krSVM.

References

1. Vapnik, V.: The Nature of Statistical Learning Theory. Springer, New York (1995)
2. Guyon, I.: Web page on svm applications (1999). <http://www.clopinet.com/isabelle/Projects/-SVM/app-list.html>
3. MacQueen, J.: Some methods for classification and analysis of multivariate observations. In: Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, University of California Press 1, pp. 281–297, January 1967
4. Asuncion, A., Newman, D.: UCI repository of machine learning databases (2007)
5. LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W., Jackel, L.: Backpropagation applied to handwritten zip code recognition. *Neural Comput.* **1**(4), 541–551 (1989)

6. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. *Proc. IEEE* **86**(11), 2278–2324 (1998)
7. van der Maaten, L.: A new benchmark dataset for handwritten character recognition (2009). http://homepage.tudelft.nl/19j49/Publications_files/characters.zip
8. Cristianini, N., Shawe-Taylor, J.: *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*. Cambridge University Press, New York (2000)
9. Platt, J.: Fast training of support vector machines using sequential minimal optimization. In: Schölkopf, B., Burges, C., Smola, A. (eds.) *Advances in Kernel Methods Support Vector Learning*, pp. 185–208. MIT Press, Cambridge (1999)
10. Do, T.-N.: Non-linear classification of massive datasets with a parallel algorithm of local support vector machines. In: Le Thi, H.A., Nguyen, N.T., Do, T.V. (eds.) *Advanced Computational Methods for Knowledge Engineering. AISC*, vol. 358, pp. 231–241. Springer, Heidelberg (2015)
11. Wu, X., Kumar, V., Quinlan, J.R., Ghosh, J., Yang, Q., Motoda, H., McLachlan, G.J., Ng, A., Liu, B., Yu, P.S., Zhou, Z.H., Steinbach, M., Hand, D.J., Steinberg, D.: Top 10 algorithms in data mining. *Knowl. Inf. Syst.* **14**(1), 1–37 (2007)
12. Chang, C.C., Lin, C.J.: LIBSVM : a library for support vector machines. *ACM Trans. Intell. Syst. Technol.* **2**(27), 1–27 (2011)
13. Vapnik, V.: Principles of risk minimization for learning theory. In: *Advances in Neural Information Processing Systems 4*, (NIPS Conference, Denver, Colorado, USA, December 2–5, 1991), pp. 831–838 (1991)
14. Bottou, L., Vapnik, V.: Local learning algorithms. *Neural Comput.* **4**(6), 888–900 (1992)
15. Vapnik, V., Bottou, L.: Local algorithms for pattern recognition and dependencies estimation. *Neural Comput.* **5**(6), 893–909 (1993)
16. Board, OpenMP Architecture Review: OpenMP application program interface version 3.0 (2008)
17. Breiman, L.: Random forests. *Mach. Learn.* **45**(1), 5–32 (2001)
18. Dietterich, T.G.: Ensemble methods in machine learning. In: Kittler, J., Roli, F. (eds.) *MCS 2000. LNCS*, vol. 1857, pp. 1–15. Springer, Heidelberg (2000)
19. Lin, C.: *A practical guide to support vector classification* (2003)
20. Yu, H., Yang, J., Han, J.: Classifying large data sets using svms with hierarchical clusters. In: *Proceedings of the ACM SIGKDD International Conference on KDD*, pp. 306–315. ACM (2003)
21. Do, T.N., Poulet, F.: Towards high dimensional data mining with boosting of psvm and visualization tools. In: *Proceedings of 6th International Conference on Enterprise Information Systems*, pp. 36–41 (2004)
22. Boser, B., Guyon, I., Vapnik, V.: An training algorithm for optimal margin classifiers. In: *Proceedings of 5th ACM Annual Workshop on Computational Learning Theory of 5th ACM Annual Workshop on Computational Learning Theory*, pp. 144–152. ACM (1992)
23. Osuna, E., Freund, R., Girosi, F.: An improved training algorithm for support vector machines. In: Principe, J., Gile, L., Morgan, N., Wilson, E. (eds.) *Neural Networks for Signal Processing VII*, pp. 276–285 (1997)
24. Mangasarian, O., Musicant, D.: Lagrangian support vector machines. *J. Mach. Learn. Res.* **1**, 161–177 (2001)
25. Fung, G., Mangasarian, O.: Proximal support vector classifiers. In: *Proceedings of the ACM SIGKDD International Conference on KDD*, pp. 77–86. ACM (2001)

26. Mangasarian, O.: A finite newton method for classification problems. Technical report 01–11, Data Mining Institute, Computer Sciences Department, University of Wisconsin (2001)
27. Suykens, J., Vandewalle, J.: Least squares support vector machines classifiers. *Neural Process. Lett.* **9**(3), 293–300 (1999)
28. Shalev-Shwartz, S., Singer, Y., Srebro, N.: Pegasos: primal estimated sub-gradient solver for SVM. In: *Proceedings of the Twenty-Fourth International Conference Machine Learning*, pp. 807–814. ACM (2007)
29. Bottou, L., Bousquet, O.: The tradeoffs of large scale learning. In: Platt, J., Koller, D., Singer, Y., Roweis, S. (eds.) *Advances in Neural Information Processing Systems*, vol. 20, pp. 161–168. NIPS Foundation (2008). <http://books.nips.cc>
30. Do, T.N., Poulet, F.: Incremental svm and visualization tools for bio-medical data mining. In: *Proceedings of Workshop on Data Mining and Text Mining in Bioinformatics*, pp. 14–19 (2003)
31. Do, T.N., Poulet, F.: Classifying one billion data with a new distributed svm algorithm. In: *Proceedings of 4th IEEE International Conference on Computer Science, Research, Innovation and Vision for the Future*, pp. 59–66. IEEE Press (2006)
32. Fung, G., Mangasarian, O.: Incremental support vector machine classification. In: *Proceedings of the 2nd SIAM International Conference on Data Mining* (2002)
33. Poulet, F., Do, T.N.: Mining very large datasets with support vector machine algorithms. In: Camp, O., Filipe, J., Hammoudi, S., Piattini, M. (eds.) *Enterprise Information Systems V*, pp. 177–184 (2004)
34. Do, T.: Parallel multiclass stochastic gradient descent algorithms for classifying million images with very-high-dimensional signatures into thousands classes. *Vietnam J. Comput. Sci.* **1**(2), 107–115 (2014)
35. Do, T.-N., Nguyen, V.-H., Poulet, F.: Speed Up SVM algorithm for massive classification tasks. In: Tang, C., Ling, C.X., Zhou, X., Cercone, N.J., Li, X. (eds.) *ADMA 2008. LNCS (LNAI)*, vol. 5139, pp. 147–157. Springer, Heidelberg (2008)
36. Do, T.N., Poulet, F.: Mining very large datasets with svm and visualization. In: *Proceedings of 7th International Conference on Enterprise Information Systems*, pp. 127–134 (2005)
37. Boley, D., Cao, D.: Training support vector machines using adaptive clustering. In: Berry, M.W., Dayal, U., Kamath, C., Skillicorn, D.B. (eds.) *Proceedings of the Fourth SIAM International Conference on Data Mining*, Lake Buena Vista, Florida, USA, April 22–24, 2004, SIAM, pp. 126–137 (2004)
38. Tong, S., Koller, D.: Support vector machine active learning with applications to text classification. In: *Proceedings of the 17th International Conference on Machine Learning*, pp. 999–1006. ACM (2000)
39. Pavlov, D., Mao, J., Dom, B.: Scaling-up support vector machines using boosting algorithm. In: *15th International Conference on Pattern Recognition*, vol. 2, pp. 219–222 (2000)
40. Do, T.N., Le-Thi, H.A.: Classifying large datasets with svm. In: *Proceedings of 4th International Conference on Computational Management Science* (2007)
41. Do, T.N., Fekete, J.D.: Large scale classification with support vector machine algorithms. In: Wani, M.A., Kantardzic, M.M., Li, T., Liu, Y., Kurgan, L.A., Ye, J., Ogihara, M., Sagiroglu, S., Chen, X.-W., Peterson, L.E., Hafeez, K. (eds.) *The Sixth International Conference on Machine Learning and Applications, ICMLA 2007*, Cincinnati, Ohio, USA, 13–15 December 2007, pp. 7–12. IEEE Computer Society (2007)

42. Freund, Y., Schapire, R.: A short introduction to boosting. *J. Jpn. Soc. Artif. Intell.* **14**(5), 771–780 (1999)
43. Breiman, L.: Arcing classifiers. *Ann. Stat.* **26**(3), 801–849 (1998)
44. Jacobs, R.A., Jordan, M.I., Nowlan, S.J., Hinton, G.E.: Adaptive mixtures of local experts. *Neural Comput.* **3**(1), 79–87 (1991)
45. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the em algorithm. *J. Roy. Stat. Soc.:Ser B* **39**(1), 1–38 (1977)
46. Vincent, P., Bengio, Y.: K-local hyperplane and convex distance nearest neighbor algorithms. In: *Advances in Neural Information Processing Systems*, pp. 985–992. The MIT Press (2001)
47. Zhang, H., Berg, A., Maire, M., Malik, J.: SVM-KNN: discriminative nearest neighbor classification for visual category recognition. In: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 2126–2136 (2006)
48. Yang, T., Kecman, V.: Adaptive local hyperplane classification. *Neurocomputing* **71**(13–15), 3001–3004 (2008)
49. Segata, N., Blanzieri, E.: Fast and scalable local kernel machines. *J. Mach. Learn. Res.* **11**, 1883–1926 (2010)
50. Cheng, H., Tan, P.N., Jin, R.: Efficient algorithm for localized support vector machine. *IEEE Trans. Knowl. Data Eng.* **22**(4), 537–549 (2010)
51. Kecman, V., Brooks, J.: Locally linear support vector machines and other local models. In: *The 2010 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–6 (2010)
52. Ladicky, L., Torr, P.H.S.: Locally linear support vector machines. In: Getoor, L., Scheffer, T., (eds.) *Proceedings of the 28th International Conference on Machine Learning, ICML 2011, Bellevue, Washington, USA, June 28 – July 2, 2011*, pp. 985–992. Omnipress (2011)
53. Gu, Q., Han, J.: Clustered support vector machines. In: *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2013, Scottsdale, AZ, USA, April 29 – May 1, 2013, JMLR Proceedings*, vol. 31, pp. 307–315 (2013)

An Efficient Document Indexing-Based Similarity Search in Large Datasets

Trong Nhan Phan^{1(✉)}, Markus Jäger¹, Stefan Nadschläger¹,
Josef Küng¹, and Tran Khanh Dang²

¹ Institute for Application Oriented Knowledge Processing,
Johannes Kepler University Linz, Linz, Austria
{nphan, mjaeger, snadschlaeger, jkueng}@faw.jku.at

² Faculty of Computer Science and Engineering,
HCMC University of Technology, Ho Chi Minh City, Vietnam
khanh@cse.hcmut.edu.vn

Abstract. In this paper, we principally devote our effort to proposing a novel MapReduce-based approach for efficient similarity search in big data. Specifically, we address the drawbacks of using inverted index in similarity search with MapReduce and then propose a simple yet efficient redundancy-free MapReduce scheme, which not only takes advantages over the baseline inverted index-based procedures but also adapts to various similarity measures and similarity searches. Additionally, we present other strategic methods in order to potentially contribute to eliminating unnecessary data and computations. Last but not least, empirical evaluations are intensively conducted with real massive datasets and Hadoop framework in the cluster of commodity machines to verify the proposed methods, whose promising results show how much beneficial they are when dealing with big data.

Keywords: Similarity search · Efficiency · Mapreduce · Large datasets · Clustering · Filtering · Redundancy-free capability · Document indexing

1 Introduction

While consecutively playing the important role in the wide scopes of applications such as duplicate detection, plagiarism exposure, recommendation systems, data cleaning, data clustering [9], to name a few, similarity search has also to cope with challenges in the era of big data by its “three Vs” characteristics as follows: (1) Volume demonstrates the large amount of data; (2) Velocity denotes the high speed of data; and (3) Variety represents the various data forms [11, 22]. The issue has gained lots of attention and effort whilst there are many studies which never stop experiencing and looking for favorable solutions [3, 6, 8, 13, 15, 16, 19–21]. Most of them, to the best of our knowledge, only concentrate on scalability by employing divide and conquer strategies on parallel mechanisms, such as MapReduce paradigm [5], to deal with enormous data. Many studies [6, 8, 10, 12–16, 19, 20], on the other hand, additionally utilize a data index structure known as an inverted index or a postings list to allow fast text searches, which is widely-used in the area of information retrieval in general and in similarity

search in particular. Nevertheless, inverted index-based methods encounter three main problems when they are performed in MapReduce paradigm as following: (1) Every key-value pair in the inverted index has to be scanned sequentially because of the full-scan manner of MapReduce as well as the structure of the inverted index; (2) Processing data from the inverted index brings much redundancy to identify candidate pairs among documents due to their duplicate values; and (3) It is not convenient to derive the total length of each document for fast set-based similarity computing, like Jaccard or Dice [18, 19] for example, in order to speed up the similarity computing process. These problems implicitly lead to complicated data processing and affect the overall performance. Motivated from finding out an efficient similarity search under the big data context, we propose a novel MapReduce-based approach, in this paper, not only to support resolving scalability but also to take care of data redundancy and intensive data-driven processing manners which originally exist in MapReduce paradigm. Other than improving the overall performance of similarity search, our goal basically aims at what various kinds of applications might benefit and facilitate from our methods. Hence, our main contributions can be generally summarized as follows:

1. We address the three common problems with which inverted index-based methods usually encounter.
2. We then propose a simple yet efficient redundancy-free MapReduce scheme, which not only overcomes the problems from the baseline inverted index-based procedures but also has its adaptability to diverse similarity measures as well as different similarity searches such as pairwise similarity, range query, and K-Nearest Neighbor (K-NN) query.
3. We consider promising strategies that contribute to eliminating dissimilar candidate pairs and unnecessary computations as well as diminishing data redundancy throughout MapReduce processes in order to improve the effectiveness of similarity search.
4. We intensively conduct empirical experiments with real massive datasets to verify our proposed methods, whose results shows how potentially beneficial the methods are when dealing with big data.

The rest of the paper is organized as follows: Sect. 2 presents state-of-the-art which are pointed out how close and different they are when compared to our research work. Section 3 introduces the general concepts related to similarity search and MapReduce as well as some definitions and notations we use in the paper. Next, the proposed clustering scheme, the redundancy-free capability, and other collaborative strategies are given in Sect. 4. Afterwards, several empirical experiments are measured and evaluated in Sect. 5 before our remarks in Sect. 6.

2 Related Work

Efficiently doing similarity search and improving performance are of the main objectives in which much work is interested and calls for much attention. Dittrich et al. [7] do research related to Hadoop efficient processing. Their aim is to improve Hadoop performance in many different ways such as partitioning data layouts and building

indices. In order to achieve the goal, they have to, however, change the Hadoop pipelines and get involved in many low-level components inside Hadoop as well as Hadoop distributed file system. Having a different approach but still towards the same objective, we approach performance improvement from the point of view of high-level layers, i.e., algorithms and schemes, such that we build indices and exploit them for candidate search during the MapReduce jobs run time. Besides, Deng et al. [6] present a three-phase MapReduce-based algorithm for string similarity joins in that the first MapReduce operation is for the filter stage and the last two MapReduce operations are for the verification stage. In the verification stage, their algorithm needs, however, to re-access the original datasets whilst our approach only accesses the datasets once from the beginning stage. Rong et al. [19] also introduce a three-phase MapReduce algorithm for string similarity join. Their objective is to reduce the number of candidate string pairs as well. In order to do that, they apply multiple prefix filtering technique, which is based on different global orderings, to their algorithm. Nevertheless, the algorithm behaves in a full-scan manner while our method performs a clustering technique which helps access the right data. Additionally, there is no mention of resolving the redundancy of string pairs as we do in our approach.

Meanwhile, Zadeh and Goel show how to assess MapReduce algorithms. According to their work in [21], the two main complexity measures for MapReduce are the largest bucket reduce because of “the curse of the last reducer” and the shuffle size because of the total file I/O. Thus, it emerges an essential need to reduce candidate sizes as much as possible throughout MapReduce processes. In order to deal with this problem, Kolb et al. [10] focus on how to eliminate redundant similarity comparison between pairs. At REDUCE task, when considering candidate pairs, the reducers only compare those which are disjoint from the list of smaller keys. In contrast to our approach, we do not attach any additional data to intermediate key-value pairs for duplicate-pairs detection at reducers. As an alternative, we keep them identically output in a natural way from mappers and then immediately derive the similarity score between a pair of document. In addition, Metwally and Faloutsos from the work [13] propose a scalable MapReduce-based framework for discovering all-pairs similarity. This method, however, suffers heavy storage and transmission costs due to redundant data in key-value pairs, which is avoided by our method. In another work engaging in diminishing unnecessary data and computations, Phan et al. [15, 16] propose MapReduce-based filtering schemes in an effort of dealing with scalability and improving similarity search with MapReduce. The schemes are shown to generally adapt to the most common similarity search cases such as pairwise similarity, pivot case, range query, and k-Nearest Neighbor query while assuring unqualified candidates are sooner discarded. Meanwhile, Lin in [12] studies three MapReduce algorithms for brute force, large-scale ad-hoc retrieval, and Cartesian product of postings lists. Nevertheless, their concern is typically about scalability aiming at the large amount of data. In the scope of this paper, we not only integrate collaborative strategic refinements to reduce the search space but also address the standard problems of the baseline inverted index-based procedures and data redundancy. Furthermore, our proposed approach easily adapts to different similarity measures thanks to the document indexing-based data structures which interchangeably denote the term as document indexes.

3 Preliminaries

3.1 Similarity Search

Consider a universal set $\Omega = \{D_1, D_2, D_3, \dots, D_n\}$, which represents a set of n documents. In the scope of this paper, we employ the concept *k-Shingles* from the work in [18, 20] instead of terms to represent a document, whose idea is that a near duplicate object can be identified by the shingles starting with stop words. Furthermore, *k-shingles* originating from natural language processing are commonly exploited to better represent documents than using terms because of their continuous order while two documents might have the same number of terms but they turn out to appear in different positions which lead to different similarity in terms of meaning. As a consequence, a document from now on is represented by a set of shingles $D_i = \{SH_1, SH_2, \dots, SH_k\}$, and the length of a document $||D_i||$ is known as the total number of shingles belonging to the document.

Definition 1 (k-Shingles). Given a document D_i as a string of characters, *k-shingles* are defined as any sub-string having the length k found in the document.

Definition 2 (Similarity Search). Given a document D_i and a similarity threshold ε , the similarity search looks for all document pairs (D_i, D_j) in the universal set Ω , such that their similarity scores $SIM(D_i, D_j) \geq \varepsilon$.

In order to derive the similarity score between a document pair, we utilize the most widely-used similarity measure known as Jaccard coefficient [13, 16, 18, 19, 21] for fast set-based similarity computing. The form of Jaccard is given below:

$$SIM(D_i, D_j) = \frac{D_i \cap D_j}{D_i \cup D_j} \quad (1)$$

The value domain of $SIM(D_i, D_j)$ is within the range $[0, 1]$. If the document D_i is more similar to the document D_j , their similarity score is close to 1. Otherwise, their similarity score is close to 0. Last but not least, in the scope of this paper, we use the sign $[\cdot]$ to demonstrate a list, the sign $[[\cdot], [\cdot]]$ to specify a list of lists.

3.2 MapReduce Paradigm

Dean and Ghemawat in [5] present MapReduce (MR) as an effective parallel programming paradigm dealing with scalability. The basic idea is to divide a big problem into smaller ones which can be easily done in parallel in a cluster of commodity machines. The main parts of MapReduce constitute a MAP function, which produces intermediate key-value pairs, and a REDUCE function, which deliver results from the key-value pairs. When the MapReduce paradigm is deployed in the cluster, one machine plays the role of master while the others take the responsibility as workers. The master dynamically assigns MapReduce tasks to free workers in the system. Those which are assigned MAP tasks are called mappers whilst those which are assigned REDUCE tasks named as reducers. The principle data flow of a MapReduce phase is

briefly described as follows: (1) Input data whose form is of key-value pairs $[key_1, value_1]$ from the distributed file system is split into m Map tasks; (2) Mappers execute MAP function and produce r local files carrying intermediate key-value pairs $[key_2, value_2]$; (3) The shuffling process is then in charge of grouping these pairs into $[key_2, [value_2]]$ according to the keys; and (4) Reducers execute REDUCE function to aggregate the key-value pairs and derive the final results which are eventually written back to the distributed file system. To avoid ambiguity, we use the term MapReduce operations when generally mentioning both of them as a whole. Otherwise, they are separately referred as MAP and REDUCE tasks. Additionally, the terms candidate pairs refer to candidate key-value pairs. In other cases, either candidate clusters or candidate document pairs are explicitly pointed out.

4 The Proposed Methods

4.1 The Clustering Scheme

Due to the fact that MapReduce paradigm itself performs a full-scan fashion to the data, it would be slow to directly work with every single shingle as a unit in order to check whether it matches to the set of query shingles. Even though an inverted index, also known as a postings list, is widely-used in information retrieval and well-employed in lots of research work [6, 8, 10, 12–16, 19, 20] to achieve fast text searches, this method has three main drawbacks, in terms of MapReduce paradigm, for application domains in general and for similarity search in particular. Firstly, when a document is popularly represented by a set of terms or shingles, they are then performed in a full-scan manner from the inverted index. Secondly, the inverted index produces redundant data throughout MapReduce operations, which we will later on give our further analysis in Sect. 4.2. Finally, it is not easy to derive the total length of each document without adding any further information, additional processing, or at least another MapReduce operation. Hence, a research concern related to the former matter emerges such that either “*Is there a way not to sequentially scan every data unit but still get data in need?*” or “*Is it possible to only access the right data from the portion of the whole?*”

To cope with these issues, one possible method comes from clustering techniques. The basic idea is that elemental objects are group into different clusters according to their preferred properties. Thus, a cluster becomes the representative of a group, or in other words, it plays the role of a pivot. Since then, pivots partition the search space into sub-spaces in that they navigate data access to the right objects. In our approach, we cluster shingles into different compartments, which is based on their own documents, so that we can decrease the number of unnecessary data-accessing times. From this point of view, we build the data structure where a document contains a set of its shingles in an incremental manner. That is to say, a document from now on is a cluster of its own shingles and plays the role of a pivot. In comparison with the inverted index, this way of clustering shingles brings three big advantages as follows: (1) To deal with the atomic full-scan from MapReduce paradigm, we model data into a two-layer data access in that the objects we firstly check in regard to a given query are clusters instead of shingles. If the query conditions are met, the shingles of the particular clusters are retrieved for further processing; (2) At the same time, we easily derive the total number

of shingles a document has to carry on in order to support length-based filtering as well as similarity computing afterwards; and (3) This method promotes REDUCE-2 task to be transparent. In other words, it makes REDUCE-2 task get rid out of its burden processing as usual while only play the role of a transmitter writing the final result to the distributed file system.

Figure 1 illustrates two candidate-identifying processes for an inverted index and a document index, respectively in the same dataset. When given a query D_q and a threshold, let us say, with 70 % similarity, the candidates when we apply the inequality 3 in Sect. 4.3 are those whose number of shingles should greater than 3. Consequently, only $D_1, D_2, D_4,$ and D_5 satisfy this filtering condition and are then combined with the query to be candidate pairs. In the case of the inverted index, the list of checking objects sequentially includes $[T, N, D_1, D_7, R, H, D_1, D_4, O, A, D_1, D_4, D_7, G]$. On the contrary, the process in the case of the document index performs the checking only on keys. Once a key is matched, it becomes a candidate. Hence, the list of checking objects for this case is much shorter and sequentially includes $[D_1, D_2, D_3, D_4, D_5, D_6, D_7]$. It is worth noting that a number of shingles in the universe set are usually so many than that of documents. For instance, 4000 Gutenberg files in Fig. 6b have 6358196 shingles in total. Therefore, the document index approach significantly reduces the number of checking objects. Last but not least, its checking process becomes independent of the number of shingles in each document.

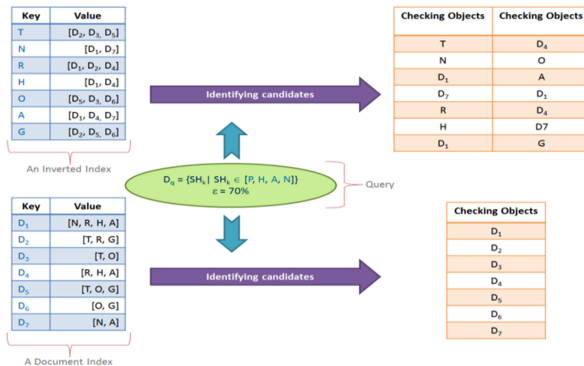


Fig. 1. Candidate-identifying processes

4.2 Redundancy-Free Compatibility

When observing data processed in MapReduce operations from the inverted index-based methods, we find out that there are lots of redundant data inner either a mapper or a reducer as well as amongst them. It is totally possible due to the fact that each mapper or reducer only processes a portion of the whole datasets. As a consequence, multiple mappers or reducers may emit duplicate key-value pairs at the same processing phase. On the other hand, because a document contains a set of shingles, or in other words, many different shingles may belong to the same document, each shingle in the inverted index carries the same information. So the research problem here is that “How to avoid redundancy throughout MapReduce operations?” In our research work,

we classify the redundancy into two classes as follows: (1) **Outer redundancy** is the case that there are at least two mappers or reducers emit the same key-value pairs; and (2) **Inner redundancy** is the case that duplicate data are emitted by only either one mapper or one reducer. Figure 2 illustrates how redundant data appear in MapReduce when the inverted index is used to search for candidate pairs. Assume that there are two mappers named $mapper_a$ and $mapper_b$, and one reducer in a MapReduce operation computing candidate similarity pairs. The inverted index, which contains references to documents D_i for each shingle represented by an upper-case letter, is fed to them as the input in MAP task. When given a query D_q , the two mappers look up the inverted index the documents sharing the same shingles with D_q . As a result, $mapper_a$ finds the candidate pairs as $[D_{q1}, D_{q4}]$ and $mapper_b$ has its candidate pairs as $[D_{q1}, D_{q4}, D_{q7}, D_{q1}, D_{q7}]$. We see that $mapper_b$ produce duplicate pairs D_{q1} and D_{q7} , which leads to the case of inner redundancy. Meanwhile, both $mapper_a$ and $mapper_b$ emit the same candidate pairs D_{q1} and D_{q4} , which gives us the case of outer redundancy. Both cases emerge very easily and frequently and add extra costs to data transmission and data computing when one works with MapReduce. Recall that other than MAP task and REDUCE task, the shuffle phase implicitly between them also suffers such a burden in the two cases.

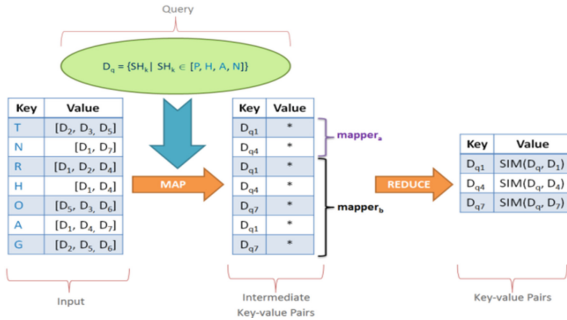


Fig. 2. Data redundancy with the inverted index

Aiming at improving performance, our proposed methods completely avoid the redundancy scenarios when seeking for candidate pairs. To keep away from the case of outer redundancy, one might look for a solution in that once mappers or reducers have emitted the pair D_{ij} , the other mappers or reducers should not emit the same pair D_{ij} . Our research work, however, does not need to do that. Actually, our methods look candidate pairs up from the clustering scheme, which is based on clusters instead of shingles themselves. As soon as there is an intersection between a pair, mappers emit it with its similarity score. Since a cluster is unique in the cluster universe, there is no chance for mappers to emit duplicate pairs. Our methods are, therefore, different from the inverted index-based ones in that shingles are not distinctive in the shingle universe due to the fact that two similar documents share the same set of shingles. Meanwhile, the case of inner redundancy impossibly takes place in our methods. The reason is that a document involving in the computing process contains distinct shingles when duplicates are sooner discarded by filtering in Sect. 4.3. In addition, the experimental

result from Fig. 9b in Sect. 5.2 shows that the collision probability of the same document is higher than that of the same shingle. In short, our methods naturally stay away from the redundancy scenarios while without adding any further information when compared to the work in [10]. Moreover, the methods easily adapt to other popular similarity searches such as pairwise similarity, range query, and K-NN query without essentially changing the scheme.

4.3 Filtering Strategies

As we know that while I/O operations are very expensive in MapReduce, useless pairs give extra-overheads not only to overall performance but also to data storage. In order to tackle this problem, we actually aim at shrinking the output from mappers. Firstly, we observe that when given a query object, a similarity search process looks for other similar ones based on their signatures, and duplicate signatures do not make sense to the similarity between a pair of objects. Moreover, it is totally redundant if we count duplicate signatures when computing similarity scores. In this paper, we use a set of shingles as the signatures of a document. We discard these duplicate shingles, therefore, from the very beginning of Map tasks, where data are at the first time read by mappers. Once the duplicates are removed, the list of shingles becomes the set of shingles, and the similarity problem turns out to be the overlap set problem [19]. Secondly, when obtaining candidate pairs, it would be useful to refine them in regard to the query object. According to a particular query, range query or K-NN query for example, we utilize the query parameters to sooner prune unnecessary candidate pairs before associating them as true similar pairs and deriving their similarity scores. More concretely, the pruning process is conducted at MAP-2 task. For instance, when given a similarity threshold ε , L_i is the length of a candidate document, and L_q is the length of a query document, the candidate pairs are the ones satisfying the below inequality, which is known as length-based filtering from the work in [18]:

$$\frac{L_i}{L_q} \geq \varepsilon \quad (2)$$

In our method, each cluster contains the number of shingles NOS . The candidate clusters should, therefore, satisfy the below inequality:

$$\|NOS\| \geq \|NOS_{query}\| * \varepsilon \quad (3)$$

On the other side, in the case of K-NN query, we simply exploit the parameter k to control the emission quantity of each mapper. This can be easily achieved if the keys in the document indexes are ordered by NOS . Consequently, we can employ this index structure to have key-value pairs in the ascending ordered manner. In other words, we will try to find those pairs having smallest NOS in order to maximize their similarity scores. Then for K-NN queries, the mappers emit the number of candidate pairs until they reach the top- k .

4.4 Examples on the Fly

Our proposed methods are packaged into two MapReduce phases. The first phase is to ahead of time prepare the data whilst the second one is to on-demand process the queries. Each phase consists of Map and Reduce tasks. In order to get insight of the proposed methods, we introduce a step-by-step example in a nutshell. Assuming that there are two different data sources where the set of documents $\{Doc_1, Doc_2, Doc_3\}$ belongs to the first one whereas the set of documents $\{Doc_4, Doc_5, Doc_6\}$ belongs to the other. As showed in Fig. 3, each document owns a set of shingles where a shingle is represented by an upper-case letter. Through MAP-1 task, mappers emit their intermediate key-value pairs of the form $[URL_i, SH_k]$. It is worth noticing that common shingles which are very popular or high-frequency shingles across the whole datasets should be, besides duplicate shingles, filtered in this phase. Common shingles can be obtained by datasets statistics or experiences. In addition, pre-defined symbols, blank space between two letters, should also be removed so that clear shingles can be easily acquired. For instance, assuming that the letter “N” is the common shingle, it should be then discarded at mappers. After MAPREDUCE-1 operation, local data are readily prepared in the form of document indexes. When given Doc_q as a query document, the same MAP-1 task and REDUCE-1 task is executed to analyze the query. These processes are put on display in Fig. 4. At MAP-2 task, only qualified candidate pairs are emitted. On the running example, the query whose NOS is equal to 5 maintains a list of its shingles $[K, O, D, E, R]$.

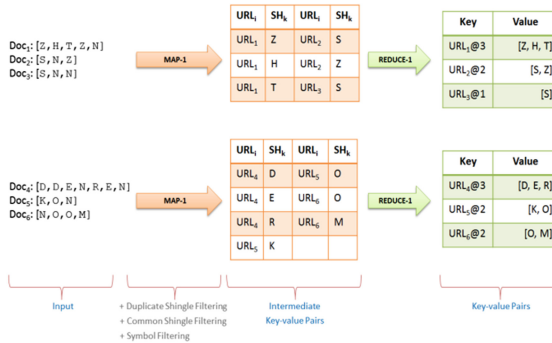


Fig. 3. MAPREDUCE-1 operation

Before finding out the intersection between the query and a document object, the length-based filtering is firstly double-checked against NOS values. More concretely, assuming that the similarity threshold ε is equal to 60 % as in Fig. 5, the candidate pairs are those satisfying the candidate pruning, i.e., the inequality 3 in Sect. 4.3. In other words, the selected pairs have to have their NOS equal or greater than 3. Consequently, none of candidates in the first local data source is taken because URL_2 and URL_3 do not satisfy the pruning condition whilst URL_1 does not have any shingles in common with the query. At the same time, only URL_4 in the second local data source is chosen for further examining. Even though the shingles of URL_5 and URL_6 in the second local data source are in the set of the query clusters, they are sooner discarded due to the

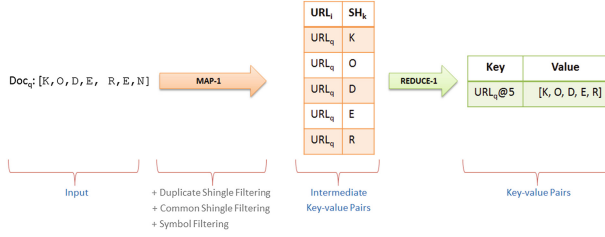


Fig. 4. MAPREDUCE-1 operation with Doc_q

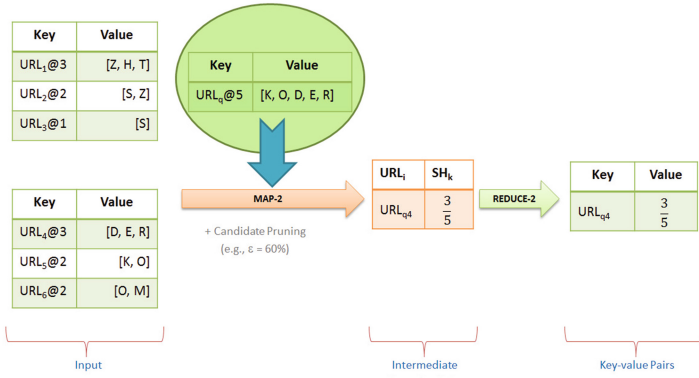


Fig. 5. MAPREDUCE-2 operation

candidate pruning. Once the candidate pairs are identified, mappers then perform similarity computing. Finally, the reducers from REDUCE-2 task output the final result. For the instance in Fig. 5, we have Doc_4 which is at least 60% similar to Doc_q with the similarity score as $3/5$. The overview of MapReduce operations and their related information are showed in Table 1. We use a special character, e.g., @, to simply illustrate the separate sub-values.

5 Empirical Experiments

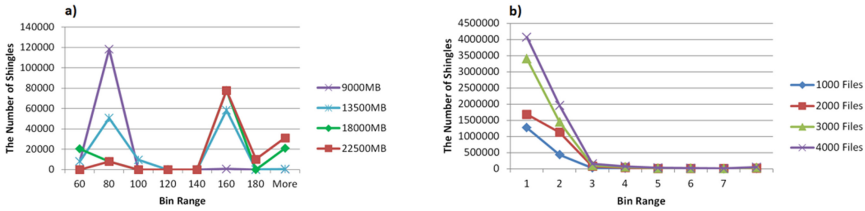
5.1 Environment Settings

To setup our experiments, we use DBLP [4] as real datasets where documents containing a number of publications are searched for their similarity. On the other side, we use other real datasets from Gutenberg Project [17], the first provider of free electronic books, to experience a large number of text files.

With DBLP Datasets. The datasets are synthetically partitioned into four packages whose sizes are exponentially increased to 9000 MB ($8 \times DBLP$), 13500 MB ($12 \times DBLP$), 18000 MB ($16 \times DBLP$), and 22500 MB ($20 \times DBLP$), respectively. Since recommended in the work [18], the size of a shingle, i.e., the K parameters for

Table 1. The overview of MapReduce operations

MapReduce	Task	Input	Output
	MAP-1	$[D_i]$	$[URL_i, SH_k]$
	REDUCE-1	$[URL_i, SH_k]$	$[URL_i@NOS_i, [SH_k]]$
	MAP-2	$[URL_i@NOS_i, [SH_k]]$	$[D_iD_j, SIM_{ij}]$
	REDUCE-2	$[D_iD_j, SIM_{ij}]$	$[D_iD_j, SIM_{ij}]$
Acronym	<ul style="list-style-type: none"> - D_i, D_j: a document object - SH_k: a k-shingle - URL_i: an uniform resource locator of D_i - NOS_i: the total number of shingles of D_i - SIM_{ij}: the similarity score between D_i and D_j - A special symbol such as “@” is used to separate the values 		

**Fig. 6.** Shingles Histograms; (a) DBLP datasets; (b) Gutenberg datasets

large documents, are chosen as 9 in DBLP datasets and as 4 in Gutenberg datasets. Figure 6a illustrates the number of shingle frequencies among the datasets. It gives a clear vision about the shingle frequency histogram with the bin range representing the interval of shingle frequency in that the majority of shingles falls into the range $[60, 100]$, $[140, 180]$, and above the range 180 while most of the shingle frequencies are from the two former ranges.

With Gutenberg Datasets. The datasets are divided into four packages separately including *1000 files*, *2000 files*, *3000 files*, and *4000 files*. These files which are randomly selected from the Gutenberg repository have their sizes ranging from 15 KB to 100 KB. In the meantime, Fig. 6b indicates the number of shingle frequencies among the Gutenberg datasets. It gives a clear vision that the majority of shingles falls under the range 4, and most of the shingle frequencies are from the range 1 to 3.

In addition, we employ the stable version 1.2.1 of Hadoop [2] as a fundamental implementation of MapReduce and deploy the Hadoop framework on the cluster of commodity machines named Alex, which has 48 nodes and 8 CPU cores and either 96 or 48 GB RAM for each node [1]. The configured capacity is set to 5 GB per node, which leads to the total 240 GB for the 48-node cluster. Besides, the number of reducers for a reduce task is set to 168. Moreover, the possible heap size of the cluster is about 629 MB, and each HDFS file has 64 MB Block Size. Last but not least, even though some parameters can be tuned or optimized to make the best fit to a particular cluster like Alex, we leave other configurations in their default mode as much as

possible due to the fact that we really want to measure the general performance with such initial settings in that whatever a cluster of commodity machines might initially have. It is worth noticing that the power of Alex is not exclusively employed for our experiments. In other words, these nodes share their computing resources to other coordinating parallel tasks in the cluster. Hence, we conduct an experiment ten times to obtain average values and their corresponding deviations. Additionally, each benchmark meets the fresh-running condition, where old benchmarks are removed before new ones start running. Furthermore, the same types of experiments are consecutively executed in order for them to have the closest running environment as much as possible. Last but not least, the benchmarks are designed to closely fit and reflex the processing capacity of the cluster.

5.2 Evaluation

In this section, we conduct our experiments with the real datasets and streaming computation models helping us pass data between MapReduce operations via the standard input and output. Figure 7 presents the performance of MapReduce operations. Figure 7a demonstrates the processing time of MR-1, MR-2, MR-Query, and the total, which turn by turn corresponds to the four DBLP dataset packages. The left vertical axis measures the average processing time of MR-1, MR-2, and MR-Query while the right vertical axis measures the average processing time of all MR-1, MR-2, and MR-Query as the whole. In general, the total costs slightly grow though the dataset sizes are doubled for each test. As we see that the cost of MR-Query is steady throughout the data packages. Besides, the cost of MR-2 does not significantly grow when the dataset size increases from 9000 MB to 22500 MB. The reason comes from the collaborative filtering where it effectively refines the candidate pairs from MR-1 and leaves the rest but small for MR-2. On the contrary, the cost of MR-1 keeps linearly rising when the dataset size keeps increasing. There are two main reasons for this. Firstly, MR-1 has to deal with enormous original data input from the very first stage, which heavily adds the processing cost. And secondly, although some filters are additionally taken, not many shingles are thrown out in comparison with the rest because of the assurance of exact similarity search. Consequently, the majority of shingles are kept for further procedures. In the meantime, Fig. 7b shows the performance of MapReduce operations on Gutenberg benchmarks. The results we get have the same trend like that on DBLP datasets. The costs of MR-Query and MR-2 seem to be stable and not much affected by the large number of files. On the contrary, the cost of MR-1 is linearly high when the number of files is increased from 1000 to 4000. It is worth noting that the cost for reducers is usually higher than that for mappers because mappers take their responsibility to tokenize the data while reducers pull intermediate key-value pairs, process them to achieve the goal, and write them into the distributed file system. Moreover, the number of mappers is usually driven by the number of distributed file system blocks in the input files whilst the number of reducers is chosen by either experiences or evaluations to a particular cluster of commodity machines. As a consequence, if the number of reducers is not suitably set, it affects the total cost in

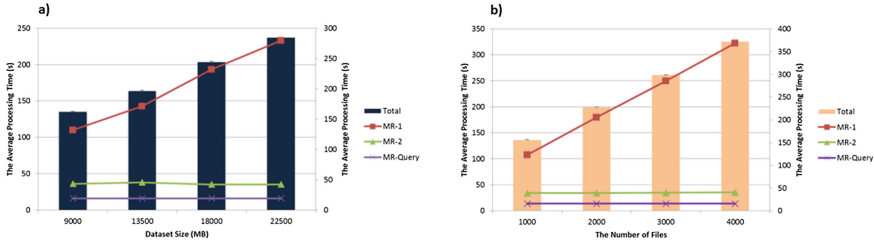


Fig. 7. Performance; (a) DBLP datasets; (b) Gutenberg datasets

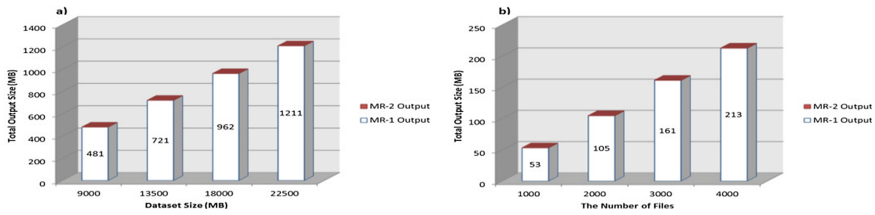


Fig. 8. Data output; (a) DBLP datasets; (b) Gutenberg datasets

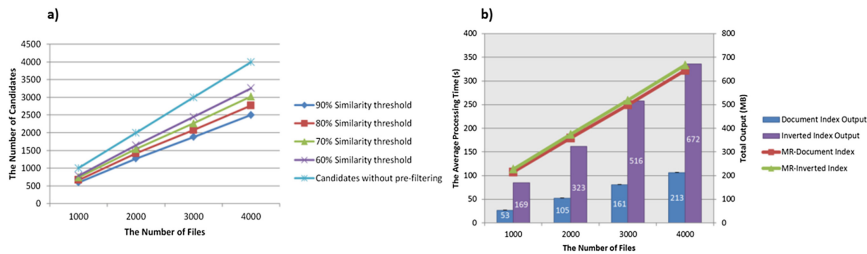


Fig. 9. Measurement; (a) Range query case; (b) The relevance between a document index and an inverted index

the end. We put, therefore, main-point processing on mappers at MAP-2 task and at the same time make reducers at REDUCE-2 task be transparent when doing similarity search, which brings an advantage to the overall performance.

On the other hand, Fig. 8 shows how much data saved from the computing processes. In an overall, the total output of MR-1 and MR-2 on DBLP datasets is much less than the input size, which accounts for 5.35 % rate of the input on the average. The total output of MR-1 and MR-2, nevertheless, accounts for 79.22 % rate of the input on the average. When the next data package in DBLP datasets is doubled, MR1-Output of this package is as nearly 1.36 times larger on the average as that of the previous package. Because of preserving as much data as possible from the datasets, the size of MR1-Output is non-trivial while that of MR2-Output is negligible, for it is around 62 KB to 161 KB. On the other hand, when the number of files is increased in Gutenberg datasets, MR1-Output of this package is as nearly 1.97 times larger on the average as that of the previous package while MR2-Output keeps its small size around

67 KB to 272 KB. Thus, the size of the entire output is totally decided by that of MR1-Output. Again, the power of filtering is completely verified at MAP-2 task. It is worth noticing that if the first MapReduce phase can be alternatively put in offline mode, the cost of the second phase is, therefore, promising in online mode. Meanwhile, Fig. 9a shows the range query case where the inequality 3 in Sect. 4.1 is applied on Gutenberg datasets. In overview, the number of candidates without filtering approximately equals to the number of input files while the number of filtered candidates is more and more when the similarity threshold increases from 60 % to 90 %. More specifically, about 37.83 % on the average unnecessary candidates are discarded in case of 90 % similarity, about 31.16 % of that number are ignored in case of 80 % similarity, about 24.39 % of that number are removed in case of 70 % similarity, and about 19.01 % of that number are filtered in case of 60 % similarity. Another experiment whose results are displayed on Fig. 9b indicates the relevance between the document index approach and the inverted index approach. Normally, the average processing time is not much different between them. The total MapReduce outputs between the two approaches significantly have, however, a big gap. The experimental result shows that building the inverted index produces approximately 3 times as many MapReduce outputs as building the document index. Hence, the document index saves more data for further processing than the inverted index.

6 Summary

In this paper, we propose a novel MapReduce-based approach for efficient similarity search. Apart from dealing with scalability, we also consider the drawbacks of the inverted index in terms of similarity search. In addition, we promote a simple yet efficient redundancy-free MapReduce scheme, which shows its advantages when compared to inverted index-based procedures. Furthermore, we present strategic methods to cope with unnecessary data and computations. Last but not least, the results from intensive empirical evaluations with massive real datasets promote the efficiency of our methods. For our future work, we identify a distributed MapReduce-based architecture to which our approach conforms in order to cope with the “three Vs” of big data. Additionally, we further evaluate our proposed methods with other state-of-the-arts as well as more empirical experiments in other popular cases of similarity search and similarity measures.

Acknowledgements. Our sincere thanks to Faruk Kujundžić, Scientific Computing, Information Management team, Johannes Kepler University Linz, for his kind support in the Alex Cluster.

References

1. Alex cluster. Available on the following website link. <http://www.jku.at/content/e213/e174/e167/e186534>. Accessed 4 Feb 2014
2. Apache Hadoop. Wiki at <http://hadoop.apache.org/docs/r1.2.1/>. Accessed 8 Mar 2014

3. Bayardo, R., Ma, Y., Srikant, R.: Scaling up all pairs similarity search. In: Proceedings of the 16th International Conference on World Wide Web, pp. 131–140 (2007)
4. DBLP data set. <http://dblp.uni-trier.de/xml/>. Accessed 8 Mar 2014
5. Dean, J., Ghemawat, S.: MapReduce: simplified data processing on large clusters. In: Proceedings of the 6th Symposium on Operating Systems Design and Implementation, USENIX Association, pp. 137–150 (2004)
6. Deng, D., Li, G., Hao, S., Wang, J., Feng J.: MassJoin: a MapReduce-based algorithm for string similarity joins. In: Proceedings of the 30th IEEE International Conference on Data Engineering, pp. 340–351 (2014)
7. Dittrich, J., Richter, S., Schuh, S.: Efficient or Hadoop: why not both? *Datenbank-Spektrum* **13**(1), 17–22 (2013)
8. Elsayed, T., Lin, J., Oard, D.W.: Pairwise document similarity in large collections with MapReduce. In: Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies, Companion Volume, pp. 265–268 (2008)
9. Han, J., Kamber, M., Pei, J.: Data mining: concepts and techniques, 3rd edn. The Morgan Kaufmann Series in Data Management Systems. Morgan Kaufmann Publishers. ISBN: 978-0123814791 (2011)
10. Kolb, L., Thor, A., Rahm, E.: Don't match twice: redundancy-free similarity computation with MapReduce. In: Proceedings of the 2nd International Workshop on Data Analytics in the Cloud (2013)
11. Letouzé, E.: Big data for development: challenges & opportunities. In: Tatevossian, A.R., Kirkpatrick, R., (eds.) UN Global Pulse, pp. 1–47 (2012)
12. Lin, J.: Brute force and indexed approaches to pairwise document similarity comparisons with MapReduce. In: Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 155–162 (2009)
13. Metwally, A., Faloutsos, C.: V-SMART-join: a scalable MapReduce framework for all-pair similarity joins of multisets and vectors. *PVLDB* **5**(8), 704–715 (2012)
14. Mika, P.: Distributed indexing for semantic search. In: Proceedings of the 3rd International Semantic Search Workshop, pp. 1–4 (2010)
15. Phan, T.N., Küng, J., Dang, T.K.: An efficient similarity search in large data collections with MapReduce. In: Dang, T.K., Wagner, R., Neuhold, E., Takizawa, M., Küng, J., Thoai, N. (eds.) FDSE 2014. LNCS, vol. 8860, pp. 44–57. Springer, Heidelberg (2014)
16. Phan, T.N., Küng, J., Dang, T.K.: An elastic approximate similarity search in very large datasets with MapReduce. In: Hameurlain, A., Dang, T.K., Morvan, F. (eds.) Globe 2014. LNCS, vol. 8648, pp. 49–60. Springer, Heidelberg (2014)
17. Project Gutenberg. <http://www.gutenberg.org/>. Accessed 8 Mar 2014
18. Rajaraman, A., Ullman J.D.: Finding similar items. In: Mining of Massive Datasets, 1st edn, pp. 71–127 (Chap. 3). Cambridge University Press, Cambridge (2011)
19. Rong, C., Lu, W., Wang, X., Du, X., Chen, Y., Tung, A.K.H.: Efficient and scalable processing of string similarity join. *IEEE TKDE* **25**(10), 2217–2230 (2013)
20. Theobald, M., Siddharth, J., Paepcke, A.: Spotsigs: robust and efficient near duplicate detection in large web collections. In: Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 563–570 (2008)

21. Zadeh, R.B., Goel, A.: Dimension independent similarity computation. *J. Mach. Learn. Res.* **14**(1), 1605–1626 (2013)
22. Zikopoulos, P.C., Eaton, C., DeRoos, D., Deutsch, T., Lapis, G.: *Understanding big data: analytics for enterprise class Hadoop and streaming data*. McGraw-Hill Osborne Media, New York. ISBN: 978-0071790536 (2012)

Using Local Rules in Random Forests of Decision Trees

Thanh-Nghi Do^(✉)

College of Information Technology, Can Tho University, Can Tho, Vietnam
dtngchi@cit.ctu.edu.vn

Abstract. We propose to use local labeling rules in random forests of decision trees for effectively classifying data. The decision rules use the majority vote for labeling at terminal-nodes in decision trees, maybe making the classical random forest algorithm degrade the classification performance. Our investigation aims at replacing the majority rules with the local ones, i.e. support vector machines to improve the prediction correctness of decision forests. The numerical test results on 8 datasets from UCI repository and 2 benchmarks of handwritten letters recognition showed that our proposal is more accurate than the classical random forest algorithm.

Keywords: Decision trees · Random forests · Labeling rules · Local rules · Support vector machines (SVM)

1 Introduction

Decision trees [1,2] are considered to be powerful and popular for supervised classification [3]. Successful applications of decision trees have been reported for such varied fields as pattern recognition, data mining and bio-informatics [4–6]. The attractiveness of tree-based algorithms is due to the fact that the decision tree model is built in the simple way (fast, very few tunable parameters), on any type of data (numerical, nominal). The decision tree model represents rules that can be easy to understand the relationships of input variables to a target one (label). This can help the data miner to avoid the risk of wrong decisions because he gets more comprehensibility and confidence in the decision model.

However, in spite of their desirable properties, decision tree models are less accurate than support vector machines (SVM [7]) or neural networks [8,9]. Since the nineties, the pioneer works proposed by Kearns and Valiant [10–12] stand out for having essentially initiated multiple research within the machine learning community, to study the strategy for boosting a weak learning algorithm (e.g. decision trees) into an accurate strong one. The main idea is to combine multiple weak classifiers into an ensemble of classifiers that performs better than single one. The success of ensemble methods is usually explained with Bias-variance framework [13–16]. Bias term is the systematic error which is independent of the learning sample. Variance term is the error due to the variability of the

model with respect to the learning sample randomness. The performance of a learning algorithm is asserted through these two key terms. And then, variations of ensemble-based learning algorithms use Bias-variance framework to improve the performance of a weak classifier. The Bagging technique proposed by Breiman [14] aims to reduce the variance of a learning algorithm without increasing its bias too much. The Boosting strategy proposed by Freund and Schapire [17] tries to simultaneously reduce the bias and the variance. Recently, the random forests approach developed by Breiman [18] is to reduce the variance of decision tree learning while keeping the low bias. Random forests algorithm achieves high accuracy compared with state-of-the-art supervised classification algorithms, including AdaBoost [17] and SVM [7].

At terminal-nodes of decision trees in habitual forests, the labeling rules are the majority vote (plurality label), maybe making the strength of the individual trees are reduced. And then, the prediction of random forests is less efficient. Our research is to replace the majority rules with the local ones, i.e. SVM [7] to improve the prediction correctness of decision trees. The numerical test results on 8 datasets from UCI repository [19] and 2 benchmarks of handwritten letters recognition [20,21] showed that our proposal is more accurate than the classical random forest algorithm.

The paper is organized as follows. Section 2 presents the random forest algorithm and our proposed local labeling rules in decision forests. Section 3 shows the experimental results. Section 4 discusses about related work. We then conclude in Sect. 5.

2 Random Forest Algorithm Using Local Labeling Rules

Decision trees [1,2] are one of the top 10 data mining algorithms [3]. A summary of the properties [4–6,22] that make tree algorithms most powerful and popular for supervised classification, is that decision trees:

- deal with both numerical and nominal variables (attributes),
- can be used as an “off-the-shelf” procedure with very few tunable parameters,
- scale for large datasets,
- handle well irrelevant input variables,
- represent rules that can be easy to understand the relationships of input variables to a target one (label).

However, the main drawback of decision trees is that they are less accurate than recently learning algorithms, including SVM [7], neural networks [8,9] or tree-based ensemble, i.e. AdaBoost [17], Bagged trees [14]. The tree-based ensemble learning tries to combine multiple decision trees to form an ensemble of trees that is more accurate than the single one. The tree-based ensembles aim at reducing bias and/or variance [14,23,24].

2.1 Random Forests

Recently, one of the most successful tree-based ensembles is the random forests proposed by Breiman [18]. It is developed from early Bagged trees of Breiman [14], the random subspace method of Ho [25] and the forest using random attribute selections of Amit and Geman [26]. The main idea of the random forests algorithm aims at training an ensemble of high performance decision trees (relatively low bias) with high diversity between individual trees¹ in the forest (hence reducing the variance). Breiman proposed to use two strategies to keep low bias and low dependence between trees in the forest. Due to the construction of low bias trees, the individual trees are built without pruning, i.e. which are grown to maximum depth. To achieve the diversity of the trees, the strategy is to use a bootstrap replica from the original training set to learn the trees and randomly choose a subset of attributes on which to base the calculation of the best split at a decision node.

Random forest algorithm is described in Algorithm 1 and Fig. 1. The classification of a new individual x is an unweighted majority vote of the resulting trees.

Algorithm 1. Random forest algorithm

```

input :
    training set  $n$  individuals and  $p$  attributes, denoted by  $D$ 
    number of trees  $t$ 

output:
    ensemble of trees  $\{DT_1, DT_2, \dots, DT_t\}$ 

1 begin
2   for  $i \leftarrow 1$  to  $t$  do
    - draw a bootstrap sample, Bootstrap -  $i$  of size  $n$  from training dataset  $D$ 
    - learning a tree  $DT_i$  from Bootstrap -  $i$ 
      for each node of the tree, randomly choose  $p'$  attributes from  $p$  attributes
      and calculate the best split among these  $p'$  attributes
      the tree is grown to its maximal depth without pruning
3   end
4   return an ensemble of trees  $\{DT_1, DT_2, \dots, DT_t\}$ 
5   classification of a new individual  $x$ :
    Majority-vote  $\{DT_1(x), DT_2(x), \dots, DT_t(x)\}$ 
6 end

```

2.2 Local Labeling Rules in Decision Forests

Let consider more details on the prediction of an individual tree in the forest, classical decision tree algorithms, including CART [1], C4.5 [2], use the majority

¹ Two classifiers are diverse if they make different errors on new datapoints [16].

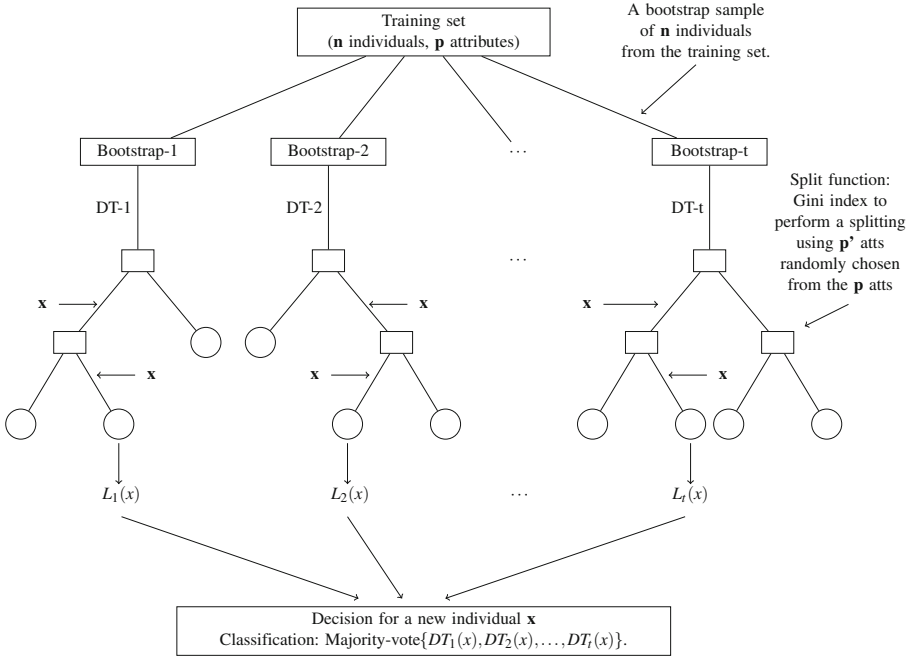


Fig. 1. Random forest algorithm

vote (plurality label) to predict the label of an individual x falling into a terminal-node (leaf node) of the tree. Figure 2 illustrates an example of the tree trained by C4.5. Two terminal-nodes with $x_2 \geq 0.495$ are not purely. The labels of individuals at the leaf-node are not the same. The majority labeling rules used in these cases maybe making the strength of the individual trees are reduced. And then, the prediction of random forests is less efficient because the classification performance of forests is dependent on the strength of individual trees.

Our proposal is to replace the majority rules with the local ones, i.e. SVM [7] to improve the prediction correctness of decision trees. For a terminal-node $Leaf_i$ with mixture of labels, a SVM model SVM_i is learnt from the individuals in this leaf node (illustrated in Fig. 3).

Furthermore, the individual trees are grown to maximum depth to achieve the low bias. It means that the minimum number of individuals is 2 (that must exist in a node in order for a split to be attempted). In [27], Vapnik points out the trade-off between the capacity of the local learning system and the number of available individuals. In this context of local labeling rules in decision trees, if the size of a terminal-node is small (i.e. 2) then the locality is extremely with a very low capacity. Therefore, the main idea is to reduce the locality and increase the capacity, thus this improves the resulting performance of local labeling SVM rules. It leads to set a large enough value (e.g. 200) to the minimum number of individuals (early stopping growing tree).

Learning SVM models

Training a binary classification model for a terminal-node with n individuals in a p attributes x_1, x_2, \dots, x_n having corresponding labels $y_i = \pm 1$, can be accomplished through the quadratic program (1).

$$\begin{aligned} \min_{\alpha} (1/2) \sum_{i=1}^n \sum_{j=1}^n y_i y_j \alpha_i \alpha_j K \langle x_i, x_j \rangle - \sum_{i=1}^n \alpha_i \\ \text{s.t.} \begin{cases} \sum_{i=1}^n y_i \alpha_i = 0 \\ 0 \leq \alpha_i \leq C \quad \forall i = 1, 2, \dots, n \end{cases} \end{aligned} \quad (1)$$

where C is a positive constant used to tune the margin and the error.

The solution of the quadratic program (1) consists of the support vectors (for which $\alpha_i > 0$), and then, they are used to construct the separating surface and the scalar b . The classification of a new data point x is based on:

$$\text{sign} \left(\sum_{i=1}^{\#SV} y_i \alpha_i K \langle x, x_i \rangle - b \right) \quad (2)$$

SVM algorithms use different kernel functions for dealing with any complex classification tasks [28]. No algorithmic changes are required from the usual kernel function K as a linear inner product other than the substitution of the kernel evaluation, including a polynomial function of degree d , a RBF (Radial Basis Function) or a sigmoid function. We can get different support vector classification models.

A binary classification SVM solver can be extended to deal with the multi-class problem (c classes, $c \geq 3$). There are two types of approaches to build the state-of-the-art multi-class SVMs from a binary SVM. The first one is considering the multi-class case in one optimization problem [29, 30]. The second one is decomposing multi-class into a series of binary SVMs, including one-versus-all [7], one-versus-one [31] and Decision Directed Acyclic Graph [32]. Recently, hierarchical methods for multi-class SVM [33, 34] start from the whole data set, hierarchically divide the data into two subsets until every subset consists of only one class.

In practice, the most popular methods are one-versus-all (ref. LIBLINEAR [35]), one-versus-one (ref. LibSVM [36]) are due to their simplicity. With c classes ($c > 2$), the one-versus-all strategy builds c different classifiers where the i^{th} classifier separates the i^{th} class from the rest. The one-versus-one strategy constructs $c(c-1)/2$ classifiers, using all the binary pairwise combinations of the c classes. The class is then predicted with a majority vote.

Prediction in the decision tree using local SVM rules

The classification of a new individual x is as follows. Run x down the tree. If x falls into a pure terminal-node, the label prediction of x is the label of this leaf node. If x falls into a mixture terminal-node, the label of x is predicted by a SVM model learnt from this leaf node.

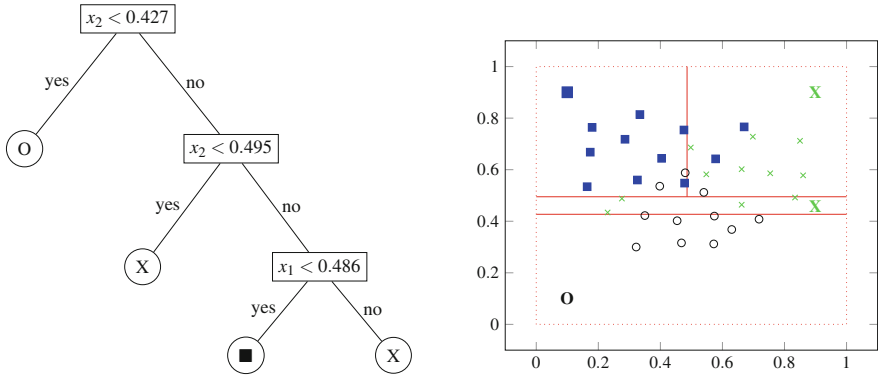


Fig. 2. Decision tree using majority labeling rules (plurality label at the terminal-nodes)

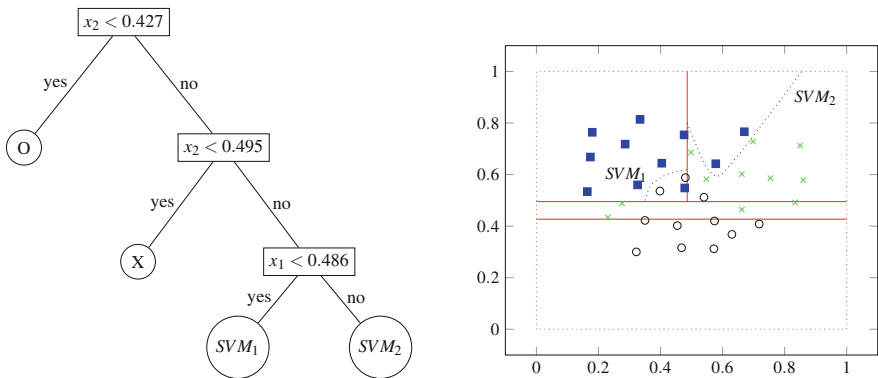


Fig. 3. Decision tree using local labeling rules (SVM classifiers at the terminal-nodes)

3 Evaluation

We are interested in the performance of the new random forest algorithm using the local labeling rules for data classification. We use the free source code of decision tree algorithm C4.5 [2] to develop the classical random forest using the majority vote (plurality label), denoted by RF-C4.5(MAJ rule) and implement our random forest with local labeling rules (SVM rules), denoted by RF-C4.5(SVM rule) using the highly efficient standard library SVM, LibSVM [36]. All tests were run under Linux on a single 2.4-GHz Pentium-4 PC with 4 GB RAM.

Our evaluation of the classification performance bases on the classification accuracy obtained by the classical RF-C4.5(MAJ rule) and our proposed RF-C4.5(SVM rule). Experiments are conducted with the 8 datasets collected from UCI repository [19] and the 2 benchmarks of handwritten letters recognition, including USPS [21], a new benchmark for handwritten character recognition

Table 1. Description of datasets

ID	Dataset	Individuals	Attributes	Classes	Evaluation protocol
1	Letter	20000	16	26	13334 Trn - 6666 Tst
2	Vowel recognition - Deterding	528	10	11	10-fold
3	Pen. Rec. of Handwritten Digits	10992	16	10	7494 Trn - 3498 Tst
4	Image Segmentation	2310	19	7	10-fold
5	Landsat Satellite	6435	36	6	4435 Trn - 2000 Tst
6	Opt. Rec. of handwritten digits	5620	64	10	3832 Trn - 1797 Tst
7	Semeion handwritten digit	1593	256	10	10-fold
8	USPS handwritten digit	9298	256	10	7291 Trn - 2007 Tst
9	Isolet	7797	617	26	6238 Trn - 1559 Tst
10	A new benchmark for Hand. Char. Rec	40133	3136	36	36000 Trn - 4133 Tst

[20]. Table 1 presents the descriptions of datasets. The evaluation protocols are illustrated in the last column of Table 1. Some data sets are already divided in training set (Trn) and testing set (Tst). For these data sets, we used the training data to build the decision forests. Then, we classified the testing set using the resulted trees. If the training set and testing set are not available then we used 10-fold cross-validation protocols to evaluate the performance. The data set is disturbed and partitioned into 10 folds. A single fold is retained as the testing set, and the remaining 9 folds are used as training set. The cross-validation process is then repeated 10 times (the folds). The 10 results from the folds can then be averaged to produce the final result.

Forest algorithms build 200 decision trees for classifying all datasets². RF-C4.5(MAJ rule) uses the parameter $minobj = 2$ (the minimum number of individuals that must exist in a node in order for a split to be attempted). RF-C4.5(SVM rule) uses the parameter $minobj = 50$ to give a trade-off between the generalization capacity [37] and the computational cost³. We propose to use RBF kernel type in SVM models because it is general and efficient [39]. We also tried to tune the hyper-parameter γ of RBF kernel (RBF kernel of two individuals $x_i, x_j, K[i, j] = \exp(-\gamma\|x_i - x_j\|^2)$) and the cost C (a trade-off between the

² We remark that we tried to vary the number of decision trees from 20 to 500 for finding the best experimental results.

³ the time complexity of learning a SVM model is in the order of n^2 where n is the number of individuals [38].

Table 2. Hyper-parameters of SVM rules at terminal-nodes in RF-C4.5(SVM rule)

ID	Datasets	γ	C
1	Letter	0.0001	100000
2	Vowel recognition - Deterding	0.01	100000
3	Pen. Rec. of handwritten digits	0.0001	100000
4	Image segmentation	0.00005	1000
5	Landsat satellite	0.001	100000
6	Opt. Rec. of handwritten digits	0.0001	100000
7	Semeion handwritten digit	0.001	100000
8	USPS handwritten digit	0.0001	100000
9	Isolet	0.0001	100000
10	A new benchmark for Hand. Char. Rec	0.0002	100000

margin size and the errors) to obtain a good accuracy. These hyper-parameters are presented in Table 2.

The accuracies of the two random forest algorithms, RF-C4.5(MAJ rule) and RF-C4.5(SVM rule) on the 10 datasets are given in Table 3 and Fig. 4. As it was expected, our RF-C4.5(SVM rule) algorithm outperforms the classical RF-C4.5(MAJ rule) on test correctness for classifying all datasets. The classical RF-C4.5(MAJ rule) achieves a mean accuracy of 94.42 %, while RF-C4.5(SVM rule) gets the best result on all 10 datasets with an average accuracy of 96.21 %, which corresponds to an improvement of 1.78 percentage points compared with RF-C4.5(MAJ rule). This superiority of RF-C4.5(SVM rule) on RF-C4.5(MAJ rule) is statistically significant, in so far as according to Wilcoxon signed rank test, the p-value of the observed results (10 wins of RF-C4.5(SVM rule) on RF-C4.5(MAJ rule) with 10 datasets) is equal to 0.001953.

In term of the computational time, the average of the training time for a forest by RF-C4.5(MAJ rule) and RF-C4.5(SVM rule) algorithms are 283.35(s) and 426.42(s), respectively. It means that RF-C4.5(MAJ rule) is 1.5 times faster than RF-C4.5(SVM rule).

4 Discussion on Related Work

Our proposal is in some aspects related to tree-based learning algorithms and local SVM models. The improvements of tree-based learning include the modification of the split function and/or the labeling rules.

The habitual tree construction uses the univariate splitting at non-terminal nodes [1, 2]. Thus, the classification performance of trees is reduced, particularly when dealing with datasets having dependencies among attributes. Due to this problem, the algorithm OC1 proposed by [40] uses multivariate splitting criteria (oblique split) -where several attributes may participate in a single node split test-, may dramatically improve the tree performance. The extensions of the OC1

Table 3. Classification results in terms of accuracy (%)

ID	Datasets	Classification accuracy(%)	
		RF-C4.5(MAJ rule)	RF-C4.5(SVM rule)
1	Letter	94.72	97.19
2	Vowel recognition - Deterding	95.76	98.65
3	Pen. Rec. of handwritten digits	96.63	98.46
4	Image segmentation	97.79	98.22
5	Landsat satellite	90.80	91.15
6	Opt. Rec. of handwritten digits	96.05	98.00
7	Semeion handwritten digit	92.14	94.21
8	USPS handwritten digit	93.77	96.11
9	Isolet	93.97	95.19
10	A new benchmark for Hand. Char. Rec	92.60	94.89
	Average	94.42	96.21

in Wu and his colleagues [41] aim at modifying the splitting criterion of the basic OC1 algorithm or post-processing OC1 output with classification SVM models [7]. Rokach and Maimon [42] illustrate that finding the good oblique split can be achieved in different ways, including linear programming proposed by [43], linear discriminant analysis [44, 45] or linear combinations of attributes [1]. Some ensemble-based methods can deal with this situation by using a large number of trees, see for example [46] and [47]). Recently random forest of oblique decision trees [48] using SVM [7], has attracted much research interests. Robnik-Sikonja proposed in [49] some possibilities for improving random forests. He investigated strategies to increase strength or to increase diversity of individual trees in the forest. The proposed forest algorithm uses several attribute evaluation measures instead of just one. The classification is based on the weighted voting.

Lazy decision tree algorithm proposed by [50] constructs the “best” decision tree for a test individual. Option tree in [51] conceptually introduces option internal-nodes to improve the prediction accuracy of decision trees.

Studies in [52–54] propose to use asymmetric entropy, off-centered entropy, Kolmogorov-Smirnov measures, respectively (instead of Shannon entropy or Gini index), as the split function in the imbalanced classification of decision trees.

For predicting the label at terminal-nodes of decision trees, the habitual labeling rules are the plurality label, maybe making the strength of the trees are reduced. The studies in [55–57] are to replace the majority label rules with the local ones, i.e. k nearest neighbors, naïve Bayes. Recently, the DTSVM algorithm [58] uses local SVM models in terminal-nodes of the tree. Ritschard and his colleagues [59] propose to use statistical implicative analysis [60] for splitting

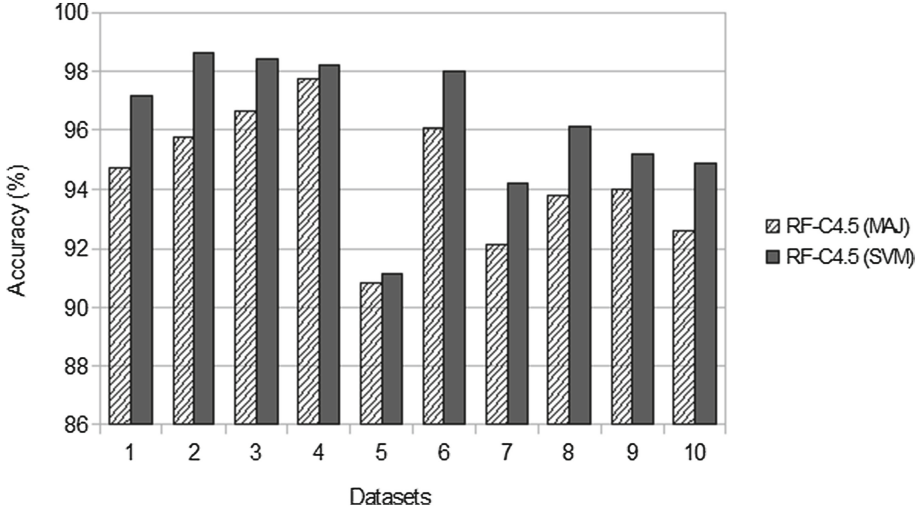


Fig. 4. Comparison of accuracy

measure and labeling rules in the decision tree learning algorithm. The OK3 algorithm [61] is the extension of tree-based learning algorithms for the regression. The main idea is to kernelize the variance measure used for evaluating the split.

Our proposal of local SVM rules in random forests is also related to local learning models. The first paper of [62] propose to use the expectation-maximization algorithm [63] for partitioning the training set into k clusters; for each cluster, a neural network is learnt to classify the individuals in the cluster. Local learning algorithms of Bottou & Vapnik [64] find k nearest neighbors of a test individual; train a neural network with only these k neighborhoods and apply the resulting network to the test individual. k -local hyperplane and convex distance nearest neighbor algorithms are also proposed in [65]. More recent local SVM algorithms include ClusterSVM [66], SVM-kNN [67], ALH [68], FaLK-SVM [69], LSVM [70], LL-SVM [71, 72], CSVN [73]. A theoretical analysis for such local algorithms discussed in [27] introduces the trade-off between the capacity of learning system and the number of available individuals. The size of the neighborhoods is used as an additional free parameters to control capacity against locality of local learning algorithms.

5 Conclusion and Future Works

This paper proposes a new random forest algorithm called RF-C4.5 (SVM rule) in which SVM models are used as local labeling rules at leaf-nodes in random forests of decision trees. No algorithmic changes are required from the classical random forest algorithm other than the modification of labeling rules at leaf-nodes. All the benefits of the original random forest method are kept. The

decision rules use local SVM models for labeling at leaf-nodes in decision trees, making the strength of the individual trees are improved. Therefore, RF-C4.5 (SVM rule) outperforms the classical random forest algorithm RF-C4.5 on test correctness for classifying all datasets from UCI repository and 2 benchmarks of handwritten letters recognition.

In the near future, we intend to provide more empirical test on large benchmarks and comparisons with other algorithms. Our proposal can be effectively parallelized. A parallel implementation that exploits the multicore processors can greatly speed up the learning and predicting tasks.

References

1. Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.: *Classification and Regression Trees*. Wadsworth International, Belmont (1984)
2. Quinlan, J.R.: *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Mateo (1993)
3. Wu, X., Kumar, V., Quinlan, J.R., Ghosh, J., Yang, Q., Motoda, H., McLachlan, G.J., Ng, A., Liu, B., Yu, P.S., Zhou, Z.H., Steinbach, M., Hand, D.J., Steinberg, D.: Top 10 algorithms in data mining. *Knowl. Inf. Syst.* **14**(1), 1–37 (2007)
4. Rokach, L., Maimon, O.Z.: *Data Mining with Decision Trees: Theory and Applications*, vol. 69. World Scientific Pub Co Inc, Singapore (2008)
5. Cutler, A., Cutler, D.R., Stevens, J.R.: Tree-based methods. In: Li, X., Xu, R. (eds.) *High-Dimensional Data Analysis in Cancer Research. Applied Bioinformatics and Biostatistics in Cancer Research*, pp. 1–19. Springer, New York (2009)
6. Berry, M.J., Linoff, G.: *Data Mining Techniques: For Marketing, Sales, and Customer Support*. Wiley, New York (2011)
7. Vapnik, V.: *The Nature of Statistical Learning Theory*. Springer, New York (1995)
8. Michie, D., Spiegelhalter, D.J., Taylor, C.C. (eds.): *Machine Learning, Neural and Statistical Classification*. Ellis Horwood, New York (1994)
9. Bishop, C.M.: *Pattern Recognition and Machine Learning. Information Science and Statistics*. Springer, New York (2006)
10. Valiant, L.: A theory of the learnable. *Commun. ACM* **27**(11), 1134–1142 (1984)
11. Kearns, M., Valiant, L.G.: Learning boolean formulae or finite automata is as hard as factoring. Technical report TR 14–88, Harvard University Aiken Computation Laboratory (1988)
12. Kearns, M., Valiant, L.: Cryptographic limitations on learning boolean formulae and finite automata. *J. ACM* **41**(1), 67–95 (1994)
13. Geman, S., Bienenstock, E., Doursat, R.: Neural networks and the bias/variance dilemma. *Neural Comput.* **4**(1), 1–58 (1992)
14. Breiman, L.: Bagging predictors. *Mach. Learn.* **24**(2), 123–140 (1996)
15. Domingos, P.: A unified bias-variance decomposition. In: *Proceedings of 17th International Conference on Machine Learning*, pp. 231–238. Morgan Kaufmann, Stanford, CA (2000)
16. Dietterich, T.G.: Ensemble methods in machine learning. In: Kittler, J., Roli, F. (eds.) *MCS 2000. LNCS*, vol. 1857, pp. 1–15. Springer, Heidelberg (2000)
17. Freund, Y., Schapire, R.: A decision-theoretic generalization of on-line learning and an application to boosting. In: *Computational Learning Theory: Proceedings of the Second European Conference*, pp. 23–37 (1995)

18. Breiman, L.: Random forests. *Mach. Learn.* **45**(1), 5–32 (2001)
19. Asuncion, A., Newman, D.: UCI repository of machine learning databases (2007)
20. van der Maaten, L.: A new benchmark dataset for handwritten character recognition (2009). http://homepage.tudelft.nl/19j49/Publications_files/characters.zip
21. LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W., Jackel, L.: Backpropagation applied to handwritten zip code recognition. *Neural Comput.* **1**(4), 541–551 (1989)
22. Hastie, T., Tibshirani, R., Friedman, J.: *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics. Springer, New York (2009)
23. Breiman, L.: Arcing classifiers. *Ann. Stat.* **26**(3), 801–849 (1998)
24. Dietterich, T.G.: An experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting, and randomization. *Mach. Learn.* **40**(2), 139–157 (2000)
25. Ho, T.K.: Random decision forest. In: *Proceedings of the Third International Conference on Document Analysis and Recognition*, pp. 278–282 (1995)
26. Amit, Y., Geman, D.: Shape quantization and recognition with randomized trees. *Mach. Learn.* **45**(1), 5–32 (2001)
27. Vapnik, V.: Principles of risk minimization for learning theory. In: Moody, J.E., Hanson, S.J., Lippmann, R.P. (eds.) *Advances in Neural Information Processing Systems*, vol. 4, pp. 831–838. Morgan Kaufmann, San Mateo (1991)
28. Cristianini, N., Shawe-Taylor, J.: *An Introduction to Support Vector Machines: And Other Kernel-based Learning Methods*. Cambridge University Press, New York (2000)
29. Weston, J., Watkins, C.: Support vector machines for multi-class pattern recognition. In: *Proceedings of the Seventh European Symposium on Artificial Neural Networks*, pp. 219–224 (1999)
30. Guermeur, Y.: *Svm multiclass, théorie et applications* (2007)
31. Kreßel, U.: Pairwise classification and support vector machines. In: Smola, A., et al. (eds.) *Advances in Kernel Methods: Support Vector Learning*, pp. 255–268. MIT Press, Cambridge (1999)
32. Platt, J., Cristianini, N., Shawe-Taylor, J.: Large margin dags for multiclass classification. In: Solla, S.A., Leen, T.K., Müller, K. (eds.) *Advances in Neural Information Processing Systems*, vol. 12, pp. 547–553. MIT Press, Cambridge (2000)
33. Vural, V., Dy, J.: A hierarchical method for multi-class support vector machines. In: *Proceedings of the Twenty-First International Conference on Machine Learning*, pp. 831–838 (2004)
34. Benabdeslem, K., Bennani, Y.: Dendrogram-based svm for multi-class classification. *J. Comput. Inf. Technol.* **14**(4), 283–289 (2006)
35. Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R., Lin, C.J.: LIBLINEAR: a library for large linear classification. *J. Mach. Learn. Res.* **9**(4), 1871–1874 (2008)
36. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines. *ACM Trans. Intell. Syst. Technol.* **2**(27), 1–27 (2011)
37. Vapnik, V., Bottou, L.: Local algorithms for pattern recognition and dependencies estimation. *Neural Comput.* **5**(6), 893–909 (1993)
38. Platt, J.: Fast training of support vector machines using sequential minimal optimization. In: Schölkopf, B., Burges, C., Smola, A. (eds.) *Advances in Kernel Methods Support Vector Learning*, pp. 185–208. MIT Press, Cambridge (1999)
39. Lin, C.: A practical guide to support vector classification (2003)

40. Murthy, S., Kasif, S., Salzberg, S., Beigel, R.: OC1: randomized induction of oblique decision trees. In: Proceedings of the Eleventh National Conference on Artificial Intelligence, pp. 322–327 (1993)
41. Wu, W., Bennett, K., Cristianini, N., Shawe-Taylor, J.: Large margin trees for induction and transduction. In: Proceedings of the Sixth International Conference on Machine Learning, pp. 474–483 (1999)
42. Rokach, L., Maimon, O.: Top-down induction of decision trees classifiers - a survey. *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.* **35**(4), 476–487 (2005)
43. Bennett, K.P., Mangasarian, O.L.: Multicategory discrimination via linear programming. *Optim. Meth. Softw.* **3**, 27–39 (1994)
44. Loh, W.Y., Vanichsetakul, N.: Tree-structured classification via generalized discriminant analysis (with discussion). *J. Am. Stat. Assoc.* **83**, 715–728 (1988)
45. Yildiz, O., Alpaydin, E.: Linear discriminant trees. *Int. J. Pattern Recogn. Artif. Intell.* **19**(3), 323–353 (2005)
46. Cutler, A., Guohua, Z.: PERT - perfect random tree ensembles. *Comput. Sci. Stat.* **33**, 490–497 (2001)
47. Geurts, P., Ernst, D., Wehenkel, L.: Extremely randomized trees. *Mach. Learn.* **63**(1), 3–42 (2006)
48. Do, T.-N., Lenca, P., Lallich, S., Pham, N.-K.: Classifying very-high-dimensional data with random forests of oblique decision trees. In: Guillet, F., Ritschard, G., Zighed, D.A., Briand, H. (eds.) *Advances in Knowledge Discovery and Management. SCI*, vol. 292, pp. 39–55. Springer, Heidelberg (2010)
49. Robnik-Sikonja, M.: Improving random forests. In: Proceedings of the Fifth European Conference on Machine Learning, pp. 359–370 (2004)
50. Friedman, J.H., Kohavi, R., Yun, Y.: Lazy decision trees. In: Proceedings of the Thirteenth National Conference on Artificial Intelligence and Eighth Innovative Applications of Artificial Intelligence Conference, AAAI 1996, IAAI 1996, vol. 1, pp. 717–724, Portland, Oregon, 4–8 Aug 1996
51. Kohavi, R., Kunz, C.: Option decision trees with majority votes. In: Proceedings of the Fourteenth International Conference on Machine Learning (ICML 1997), pp. 161–169, Nashville, Tennessee, USA, 8–12 Jul 1997
52. Marcellin, S., Zighed, D., Ritschard, G.: An asymmetric entropy measure for decision trees. In: *IPMU 2006*, Paris, France, pp. 1292–1299 (2006)
53. Lenca, P., Lallich, S., Do, T.-N., Pham, N.-K.: A comparison of different off-centered entropies to deal with class imbalance for decision trees. In: Washio, T., Suzuki, E., Ting, K.M., Inokuchi, A. (eds.) *PAKDD 2008. LNCS (LNAI)*, vol. 5012, pp. 634–643. Springer, Heidelberg (2008)
54. Do, T., Lenca, P., Lallich, S.: Enhancing network intrusion classification through the kolmogorov-smirnov splitting criterion. In: *ICTACS 2010*, pp. 50–61, Vietnam (2010)
55. Kohavi, R.: Scaling up the accuracy of naive-bayes classifiers: a decision-tree hybrid. In: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96), pp. 202–207, Portland, Oregon, USA (1996)
56. Seewald, A.K., Petrak, J., Widmer, G.: Hybrid decision tree learners with alternative leaf classifiers: an empirical study. In: *International Florida Artificial Intelligence Research Society Conference*, pp. 407–411 (2000)
57. Pham, N.K., Do, T.N., Lenca, P., Lallich, S.: Using local node information in decision trees: coupling a local decision rule with an off-centered. In: *International Conference on Data Mining*, pp. 117–123, Las Vegas, Nevada, USA, CSREA Press (2008)

58. Chang, F., Guo, C.Y., Lin, X.R., Lu, C.J.: Tree decomposition for large-scale SVM problems. *J. Mach. Learn. Res.* **11**, 2935–2972 (2010)
59. Ritschard, G., Marcellin, S., Zighed, D.A.: Arbre de décision pour données déséquilibrées : sur la complémentarité de l'intensité d'implication et de l'entropie décentrée. In: *Analyse Statistique Implicative - Une méthode d'analyse de données pour la recherche de causalités*, pp. 207–222 (2009)
60. Lerman, I., Gras, R., Rostam, H.: Elaboration et évaluation d'un indice d'implication pour données binaires. *Math. Sci. Hum.* **74**, 5–35 (1981)
61. Geurts, P., Wehenkel, L., d'Alché Buc, F.: Kernelizing the output of tree-based methods. In: Cohen, W.W., Moore, A., (eds.) *Proceedings of the Twenty-Third International Conference (ICML 2006)*, Pittsburgh, Pennsylvania, USA, 25–29 Jun 2006, *ACM International Conference Proceeding Series*, vol. 148, pp. 345–352, ACM (2006)
62. Jacobs, R.A., Jordan, M.I., Nowlan, S.J., Hinton, G.E.: Adaptive mixtures of local experts. *Neural Comput.* **3**(1), 79–87 (1991)
63. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the em algorithm. *J. Roy. Stat. Soc. B* **39**(1), 1–38 (1977)
64. Bottou, L., Vapnik, V.: Local learning algorithms. *Neural Comput.* **4**(6), 888–900 (1992)
65. Vincent, P., Bengio, Y.: K-local hyperplane and convex distance nearest neighbor algorithms. In: Dietterich, T.G., Becker, S., Ghahramani, Z. (eds.) *Advances in Neural Information Processing Systems*, vol. 14, pp. 985–992. The MIT Press, Cambridge (2001)
66. Boley, D., Cao, D.: Training support vector machines using adaptive clustering. In: Berry, M.W., Dayal, U., Kamath, C., Skillicorn, D.B. (eds.) *Proceedings of the Fourth SIAM International Conference on Data Mining*, pp. 126–137, Lake Buena Vista, Florida, USA, 22–24 Apr 2004, SIAM (2004)
67. Zhang, H., Berg, A., Maire, M., Malik, J.: SVM-KNN: discriminative nearest neighbor classification for visual category recognition. In: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 2126–2136 (2006)
68. Yang, T., Kecman, V.: Adaptive local hyperplane classification. *Neurocomputing* **71**(1315), 3001–3004 (2008)
69. Segata, N., Blanzieri, E.: Fast and scalable local kernel machines. *J. Mach. Learn. Res.* **11**, 1883–1926 (2010)
70. Cheng, H., Tan, P.N., Jin, R.: Efficient algorithm for localized support vector machine. *IEEE Trans. Knowl. Data Eng.* **22**(4), 537–549 (2010)
71. Kecman, V., Brooks, J.: Locally linear support vector machines and other local models. In: *The 2010 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–6 (2010)
72. Ladicky, L., Torr, P.H.S.: Locally linear support vector machines. In: Getoor, L., Scheffer, T., (eds.) *Proceedings of the 28th International Conference on Machine Learning, ICML 2011*, pp. 985–992, Bellevue, Washington, USA, Jun 28–Jul 2 2011, Omnipress (2011)
73. Gu, Q., Han, J.: Clustered support vector machines. In: *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2013*, Scottsdale, AZ, USA, Apr 29–May 1 2013, *JMLR Proceedings*, vol. 31, pp. 307–315 (2013)

A Term Weighting Scheme Approach for Vietnamese Text Classification

Vu Thanh Nguyen^{1(✉)}, Nguyen Tri Hai¹, Nguyen Hoang Nghia¹,
and Tuan Dinh Le²

¹ University of Information Technology VNU-HCM,
Ho Chi Minh City, Vietnam
nguyenvt@uit.edu.vn,
{11520094, 11520603}@gm.uit.edu.vn

² Long An University of Economics and Industry,
Tan An City, Long An Province, Vietnam
le.tuan@daihoclongan.edu.vn

Abstract. The term weighting scheme, which is used to convert the documents to vectors in the term space, is a vital step in automatic text categorization. The previous studies showed that term weighting schemes dominate the performance. There have been extensive studies on term weighting for English text classification. However, not many works have been studied on Vietnamese text classification. In this paper, we proposed a term weighting scheme called $normalize(tf.rf_{max})$, which is based on $tf.rf$ term weighting scheme – one of the most effective term weighting schemes to date. We conducted experiments to compare our proposed $normalize(tf.rf_{max})$ term weighting scheme to $tf.rf$ and $tf.idf$ on Vietnamese text classification benchmark. The results showed that our proposed term weighting scheme can achieve about 3 %–5 % accuracy better than other term weighting schemes.

Keywords: Term weighting scheme · Vietnamese text classification · $tf.idf$ · $tf.rf$

1 Introduction

Text classification (TC – a.k.a. text categorization) is the task of automatically sorting a set of documents into categories (or classes, or topics) from a predefined set. Text classification, falls at the crossroads of information retrieval (IR) and machine learning (ML), has drawn significant interests in the last ten years from research communities due to the rapid growth of online information. Text classification has been used in many applications such as classifying news by subjects or new groups, sorting and filtering email messages, guiding users to search through hypertext, etc.

In recent years, most of the works on English text classification have been focused on improvement of document representation models when transforming the content of textual documents into document vectors to be classified by a computer [3, 8, 10]. The common way of term weighting is *term frequency - inverse document frequency* ($tf.idf$), which was proposed in the information retrieval field [10]. It was based on the intuition that the importance of a term to a document is dependent on its frequency as well as the degree of rareness at the document level in the corpus.

In [3], they proposed supervised term weighting methods that used the known categorical information in the training corpus. They adopted the values of the three feature selections (i.e., χ^2 , information gain, and gain ratio) to substitute *idf* factor during weighting terms. Their thorough experiments did not exhibit a uniform superiority with respect to standard *tf.idf* [2]. The work in [8] showed that the traditional *tf.idf* weighting method might lose the ability of a term to discriminate the positive documents from the negative ones. Therefore, they proposed the new weighting method, namely *term frequency – relevance frequency (tf.rf)* with new factor *rf* (relevance frequency) to improve the term’s discriminating power. A disadvantage of this method is that it is suitable only for binary classifiers.

There are not many works on term weighting scheme for Vietnamese text classification. Most of the current works for Vietnamese text classification focus on performance of text classification algorithms [4], but not on the term weighting scheme.

In this paper, we proposed and evaluated the new term weighting scheme called *normalize(tf.rf_{max})* for Vietnamese text classification. Unlike the *tf.rf* term weighting scheme, our proposed scheme requires a single *rf* value for each term for multi-class problems and uses the $(1 + \log(tf))$ instead of classical *tf*. Our experimental results show that on average our scheme is better than *tf.rf* and *tf.idf* of accuracy about 3 %–5 %.

This paper is organized as follows. Section 2 describes the background and the related works. Section 3 describes our proposed term weighting scheme. Section 4 describes our experimental results and we conclude the paper in Sect. 5.

2 Background and Related Works

2.1 Related Works

Traditional Term Weighting Methods. Generally, the traditional term weighting methods are from the information retrieval field and belonged to the unsupervised term weighting methods. The simplest *binary* term weighting method assigns 1 to all terms in a document in the vector representation phase. The most widely used term weighting approaches in this group is *tf.idf (Term Frequency - Inverse Document Frequency)* [10], puts weighting to a term based on its inverse document frequency. It means that if the more documents the term appears, the less important the term is, and the weighting will be less. It can be depicted as this:

$$tf.idf = tf_{ij} * \log\left(\frac{N}{n_j}\right) \quad (1)$$

In (1), tf_{ij} represents the term frequency of term j in document i , N represents the total number of documents in the dataset, n_j represents the number of documents that term i appears. *tf* has various variants which use the logarithm operation such as $\log(tf)$, $\log(1 + tf)$, $1 + \log(tf)$ [9]. The *tf.idf* is the most famous IR field. However, the TC task differs from the IR task. For TC task, the categorical information of terms in training documents is available in advance. The categorical information is of importance for TC

task. The supervised term weighting methods used the known categorical information in training corpus.

Supervised Term Weighting Methods. The supervised term weighting methods use the prior information about the membership of training documents in predefined categories to assign weights to terms [3, 8]. One way to use this known information is to combine tf and a feature selection metric such as χ^2 , Information Gain, Gain Ratio [2, 3]. $tf.rf$ is the supervised term weighting method that combines tf and rf (*relevance frequency*) factor which proposed by Lei and Venu [8]. As mentioned in Introduction, OneVsAll transforms a multi-class classification problem into N binary classification problems, each relates to a category which is tagged as the positive category and all other categories in the training set are grouped into the negative category. Each term t requires one rf value in each category C_i , and this value is computed as follows:

$$rf = \log \left(2 + \frac{a}{\max(c, 1)} \right) \quad (2)$$

When combined with tf by a multiplication operation, the weight of term t_i is defined as:

$$tf.rf = tf * \log \left(2 + \frac{a}{\max(c, 1)} \right) \quad (3)$$

where, a is the number of documents in category C_i which contains t and c is the number of documents not in category C_i which contains t . The purpose of rf is to give more weight to terms which help classify documents into the positive category.

However, $tf.rf$ has a common shortcoming, which is the simplification of a multiclass classification problem into multiple independent binary classification problems. During the process of the simplification, the distribution of a term among categories disappeared because there are only positive category and negative category. Our scheme will deal with this problem, which requires a single rf value for each term for multi-class problem.

2.2 Background

Support Vector Machines Classifier. The simplest SVM is a binary classifier, which is mapping to a class and can identify an instance belonging to the class or not. To produce a SVM classifier for class C, the SVM must be given a set of training samples including positive and negative samples. Positive samples belong to C and negative samples do not. After text preprocessing, all samples can be translated to n-dimensional vectors. SVM tries to find a separating hyper-plane with maximum margin to separate the positive and negative examples from the training samples.

Basically, there are two types of approaches for multi-class SVM. The first one is to consider all data in one optimization (crammer and singer). The other is to decompose

multiclass into a series of binary SVMs. Although more sophisticated approaches for multi-class SVM exist, the studies in [5] have shown that OVO and DDAG are among the most suitable methods for practical use.

The OVO method is chosen to be our classification system. The OVO is constructed by training binary SVMs between pairwise classes. Thus, the OVO consists of $K(K - 1)/2$ binary SVMs for K -class problem. Each of the $K(K - 1)/2$ SVMs casts one vote for its favored class, and finally the class with most votes wins.

Feature Extraction. Feature extraction is the first step of preprocessing which is used to present the text documents into clear word format. The common steps taken for the feature extraction of Vietnamese text are:

- **Tokenization:** In Vietnamese language, boundaries between words are not spaces as in English because Vietnamese is an isolating language [4]. We use state-of-the-art word segmentation program in [6] as a tokenizer. All documents are segmented into words or tokens that will be the inputs for next steps.
- **Removing stop words:** Stop words such as “à”, “và”, “tùy”, “củ”... etc. are frequently occurring, so the insignificant words need to be removed. For this purpose a list of function words is prepared and used in the preprocessing phase as a stop list (about ~ 1000 words, collected manually).

The result puts into feature selection step to choose a subset of high discriminative features and eliminate the non-discriminative features.

Feature Selection. In text classification, as usual, Feature Selection is used to reduce the text data dimensionality. The Information Gain (IG) and Chi-square statistic (CHI) are two of the most efficient feature selections, and Document Frequency (DF) is comparable to the performance of IG and CHI [11]. In our system, we used Information Gain (IG), which is often used as a criterion in the field of machine learning. Information Gain (IG) is often used as a criterion in the field of machine learning. The Information Gain of a given feature t_k with respect to the class c_i is the reduction in uncertainty about the value of c_i when we know the value of t_k . The larger Information Gain of a feature is, the more important the feature is for categorization. Information Gain of a feature t_k toward a category c_i can be defined as follows:

$$IG(t_k, c_i) = \sum_{c \in \{c_i, \bar{c}_i\}} \sum_{t \in \{t_k, \bar{t}_k\}} P(t, c) \log \frac{P(t, c)}{P(t)P(c)} \quad (4)$$

where $P(c)$ is the fraction of the documents in category c over the total number of documents, $P(t, c)$ is the fraction of documents in the category c that contain the word t over the total number of documents. $P(t)$ is the fraction of the documents containing the term t over the total number of documents.

3 Our Proposed Term Weighting Method

As mentioned previously, for each term in a multi-class classification problem, $tf.rf$ uses N rf values, each of them for a different binary classifier. Meanwhile, our scheme, which also uses OneVsAll method and assigns single rf_{max} (maximum of all rf) to each term for all binary classifiers. Thus, the weight of the term is corresponded to the category which it represents the most. We defined $rf(C_i)$, as described in Eq. (3), where N is the total number of categories and rf_{max} is defined as followings:

$$rf_{max} = \max_{i=1 \rightarrow N} \{rf(C_i)\} \quad (5)$$

The consequence the highest value use is that our scheme is simpler than $tf.rf$. For a N -class problem, our scheme needs only one presentation for all N binary classifiers. Moreover, it has been in shown related work [8] that $tf.rf$ has the lower result than rf in some cases, which is mainly caused by the repeated noisy terms in a document. Therefore the combination of rf_{max} and $(1 + \log(tf))$ instead of tf . The $(1 + \log(tf))$ scales down the effect of noisy terms can improve the classification results.

To eliminate the length effect, we use the cosine normalization [9] to limit the term weighting range within $(0, 1)$. Specially, the binary feature representation does not use normalization since the original value is 0 or 1. Assuming that w_{ij} represents the weight of term t_i in document d_j , the final term weight w_{ij} might then be defined as:

$$normalize\ w_{ij} = \frac{w_{ij}}{\sqrt{\sum_i (w_{ij}^2)}} \quad (6)$$

To sum up, our proposed term weighting method computed weight for the term t_{ij} as following:

$$w_{ij} = normalize(tf.rf_{max}) = \frac{(1 + \log(tf)) * \max_{i=1 \rightarrow N} \{rf(C_i)\}}{\sqrt{\sum_{i=1}^n \left((1 + \log(tf)) * \max_{i=1 \rightarrow N} \{rf(C_i)\} \right)^2}} \quad (7)$$

where tf_{ij} is the frequency of t_{ij} , N is the total number of categories, $rf(C_i)$ is defined in Eq. (3), n is the total unique words (features) appearing in document j .

4 Experiments

4.1 Experimental Setup

In our experiment, SVM with OAO multi-class method was employed as our classifier because the effectiveness of the SVM has been studied in many related works and is able to deal with large dimensions of feature space [5, 7]. We used IG feature selection method to choose a subset of high discriminative features and eliminate the

non-discriminative features. We evaluated the performance of term weighting methods: *tfidf*, *tfidf*, *normalize(tf.rf_{max})* with various number of features.

The SVM tool, LIBSVM [1], developed by Chih-Jen Lin, was adopted. In the SVM training model, we selected the parameters as following: C = 1; kernel-function = linear; SVM-type = CSVM; other default parameters.

In our experiments, we used the Vietnamese corpus given in [4]. The corpus has been collected from the four largest Vietnamese online newspapers. This corpus consists of 110,583 documents and is divided into two levels. The Level 1 corpus includes the top 10 popular categories from the websites of the electronic newspapers. This corpus contains 33,759 documents for the training and 50,373 documents for the testing. The resulting vocabulary has 143,178 unique words (features). The Level 2 corpus includes 27 topics, which are the child topics of those in the Level 1 corpus. This corpus contains 14,375 documents for training and 12,076 documents for testing. The resulting vocabulary has 91,466 unique words (features).

By using IG for feature selection, the top $p \in \{1\%, 2\%, 3\%, 4\%, 5\%\}$ features are tried for each Level corpus.

4.2 Evaluation Methodology

There are different metrics used in evaluating effectiveness of document classification. In our experiment, we use the well-known accuracy metric, which is widely used in Information Retrieval System [5].

$$Accuracy = \frac{\#correctly\ predicted\ text}{\#total\ testing\ text} \times 100\ \%$$

4.3 Results

The experimental results of three term weighting methods with respect to *accuracy (%)* measure on two Level corpuses reported from Tables 1 to 2. Each line in the table shows the performance of each term weighting method at different of feature selection levels.

Level 1 Corpus. Table 1 shows the results with respect to the accuracy on the Level 1 corpus. All term weighting schemes reached a maximum of accuracy at the 5 % total

Table 1. Accuracy (%) of three term weighting schemes on Level 1

% features	<i>tf.idf</i>	<i>tf.rf</i>	<i>Normalize (tf.rf_{max})</i>
1	87.15 %	87.61 %	91.26 %
2	88.73 %	88.75 %	92.49 %
3	89.50 %	89.54 %	92.85 %
4	89.93 %	89.82 %	93.06 %
5	90.22 %	89.87 %	93.12 %
Avg.	89.11 %	89.12 %	92.56 %

features. Among these, the best accuracy 93.12 % was reached at 5 % features achieved by our proposed scheme $normalize(tf.rf_{max})$. The $normalize(tf.rf_{max})$ scheme has always been shown significant better performance than others when the percent of features increases from 1 % to 5 % total features. Particularly, the average accuracy score achieved $normalize(tf.rf_{max})$ is about 3.45 % higher than $tf.idf$ and about 3.44 % than $tf.rf$ (92.56 % for $normalize(tf.rf_{max})$, 89.12 % for $tf.rf$, 89.11 % for $tf.idf$, respectively).

Level 2 Corpus. Table 2 shows the results with respect to accuracy on the Level 2 corpus. Similar to the results of the Level 1 corpus, the achieved accuracy of all term weighting schemes on the Level 2 corpus is higher when the percent of features increases from 1 % to 5 % total features. The best accuracy value of 92.49 % was also achieved by our proposed scheme $normalize(tf.rf_{max})$ at a features size of 5 % total features. According to the Table 2, we can see that the average accuracy score of $normalize(tf.rf_{max})$ is 91.59 %, it's higher than $tf.idf$ (86.50 %) about 5.09 %, higher than $tf.rf$ (86.53 %) about 5.06 %.

Table 2. Accuracy (%) of three term weighting schemes on Level 2

% features	$tf.idf$	$tf.rf$	$Normalize(tf.rf_{max})$
1	84.56 %	84.31 %	89.93 %
2	86.01 %	86.05 %	91.32 %
3	86.95 %	86.98 %	91.99 %
4	87.03 %	87.36 %	92.24 %
5	87.97 %	87.93 %	92.49 %
Avg.	86.50 %	86.53 %	91.59 %

Therefore, the results show that our proposed scheme $normalize(tf.rf_{max})$ is better than the others in the two different Vietnamese data sets based on different category distributions. Both of the best accuracy were achieved by our proposed $normalize(tf.rf_{max})$ scheme on the two levels of data sets. This result verifies that the rf (relevance frequency) improves the term's discriminating power for text classification in $tf.rf$ and $normalize(tf.rf_{max})$. This result also verifies that the rf_{max} improves the performance $normalize(tf.rf_{max})$, which is higher than $tf.rf$ about accuracy.

5 Conclusion

In this paper, we have proposed an the term weighting scheme $normalize(tf.rf_{max})$ that employs the two improvements to $tf.rf$ – one of the best term weighting schemes to date. Different from other schemes, our scheme requires a single rf value for each term while $tf.rf$ requires many rf values in a the multi-class classification problems. Second, our scheme used the $(1 + \log(tf))$ instead of *classical* tf . The experimental results showed that our term weighting scheme can achieve 3 %–5 % of accuracy higher than $tf.rf$ and $tf.idf$ on two Vietnamese data sets with the different category distribution.

Our future work will investigate additional classifiers (for example kNN) as well as text corpuses to further validate $normalize(tf.rf_{max})$.

Acknowledgment. This research is funded by Vietnam National University, Ho Chi Minh City (VNU-HCM) under grant number C2014-26-04.

References

1. Chang, C.C., Chih, J.L.: LIBSVM: a library for support vector machines. *ACM Trans. Intell. Syst. Technol. (TIST)* **2**(3), 27 (2011)
2. Debole, F., Fabrizio, S.: Supervised term weighting for automated text categorization. In: Sirmakessis, S. (ed.) *Text Mining and Its Applications*, pp. 81–97. Springer, Berlin, Heidelberg (2004)
3. Deng, Z.-H., Tang, S.-W., Yang, D.-Q., Li, M.Z.L.-Y., Xie, K.-Q.: A comparative study on feature weight in text categorization. In: Yu, J.X., Lin, X., Lu, H., Zhang, Y. (eds.) *APWeb 2004. LNCS*, vol. 3007, pp. 588–597. Springer, Heidelberg (2004)
4. Hoang, V.C.D., et al.: A comparative study on Vietnamese text classification methods. In: 2007 IEEE International Conference on Research, Innovation and Vision for the Future. *IEEE* (2007)
5. Hsu, C.W., Chih, J.L.: A comparison of methods for multiclass support vector machines. *IEEE Trans. Neural Netw.* **13**(2), 415–425 (2002)
6. Phuong, L.H., Huyên, N.T.M., Roussanaly, A., Vinh, H.T.: A hybrid approach to word segmentation of vietnamese texts. In: Martín-Vide, C., Otto, F., Fernau, H. (eds.) *LATA 2008. LNCS*, vol. 5196, pp. 240–249. Springer, Heidelberg (2008)
7. Joachims, T.: Text categorization with support vector machines: learning with many relevant features. In: Nédellec, C., Rouveirol, C. (eds.) *ECML 1998. LNCS*, vol. 1398, pp. 137–142. Springer, Berlin, Heidelberg (1998)
8. Lei, H., Govindaraju, V.: Half-against-half multi-class support vector machines. In: Oza, N. C., Polikar, R., Kittler, J., Roli, F. (eds.) *MCS 2005. LNCS*, vol. 3541, pp. 156–164. Springer, Heidelberg (2005)
9. Leopold, E., Jörg, K.: Text categorization with support vector machines. How to represent texts in input space? *Mach. Learn.* **46**(1–3), 423–444 (2002)
10. Salton, G., Buckley, C.: Term-weighting approaches in automatic text retrieval. *Inf. Process. Manage.* **24**(5), 513–523 (1988)
11. Yang, Y., Jan, O.P.: A comparative study on feature selection in text categorization. In: *ICML*, vol. 97 (1997)

Security and Privacy Engineering

Fault Data Analytics Using Decision Tree for Fault Detection

Ha Manh Tran^(✉), Sinh Van Nguyen, Son Thanh Le, and Quy Tran Vu

Computer Science and Engineering,
International University - Vietnam National University,
Ho Chi Minh City, Vietnam
{tmha,nvsinh,ltson,vtquy}@hcmiu.edu.vn

Abstract. Monitoring events on communication and computing systems becomes more and more challenging due to the increasing complexity and diversity of these systems. Several supporting tools have been created to assist system administrators in monitoring an enormous number of events daily. The main function of these tools is to filter as many as possible events and present non-trivial events to the administrators for fault analysis and detection. However, non-trivial events never decrease on large systems, such as cloud computing systems, while investigating events is time consuming. This paper proposes an approach for evaluating the severity level of an event using a classification and regression decision tree. The approach aims to build a decision tree based on the features of old events, then use this tree to decide the severity level of new events. The administrators take advantages of this decision to determine proper actions for the non-trivial events. We have implemented and experimented the approach for software bug datasets obtained from bug tracking systems. The experimental results reveal that the accuracy scores for different decision trees are above 70 % and some detailed analyses are provided.

Keywords: Event monitoring · Fault data analytics · Fault detection · CART decision tree · Software bug report

1 Introduction

The increasing complexity and diversity of communication and computing systems makes management operations more and more challenging. Cloud computing systems [1], as an example, facilitate computing resource management operations on large computing systems to provision infrastructures, platforms and software as services. Armbrust [2] has specified 10 hindrances for managing cloud systems and services. Several hindrances including service availability, performance unpredictability and failure control are closely involved with event monitoring, one of the main functions of fault management. Monitoring events on these systems usually deals with a large number of events. The system administrators needs the support of tools that filter out many events and keep non-trivial events. However, these systems provide so many non-trivial events that

the administrators cannot handle. Furthermore, there is no guarantee that trivial events cannot cause system failure, e.g., warning events can become serious problems if there is no a proper action.

We have proposed an approach for evaluating the severity level of log events using classification and regression decision trees (CART trees). The idea of this approach is to determine the severity level of events automatically, thus providing the system administrators a decision whether further actions are needed for fault detection. The approach focuses on constructing a decision tree based on the features of old events and then using this tree to decide the severity level of new events. We have used software bug datasets obtained from bug tracking systems (BTSs) to implement and experiment the decision trees. The contribution is thus threefold:

1. Proposing an approach of using the CART decision tree for fault data analytics
2. Applying this approach to software bug datasets for evaluating the severity level of bug reports
3. Providing the performance evaluation of the approach on various software bug datasets

The rest of the paper is structured as follows: the next section includes some analysis techniques applied to software maintenance, system failure and reliability, some background of classification and regression trees in data analysis. Section 3 describes the fundamentals of growing decision trees based on classification and regression trees, focusing on entropy splitting rule and tree growing process. Some mathematical formulas and explanations are referred from the study of Breiman et al. [3]. Section 4 presents characteristics of fault data and several processes of building decision trees for fault datasets. Several experiments in Sect. 5 report the performance and efficiency of fault data analysis before the paper is concluded in Sect. 6.

2 Related Work

The authors of the study [4] have proposed an approach for analysing fault cases in communication systems. The approach exploits the characteristics of semi-structured fault data by using multiple field-value and semantic vectors for fault representation and evaluation. Note that a fault case usually contains administrative field-value and problem description parts. The approach encounters the problem of high computation cost when processing semantic matrices for large fault datasets. Another study [5] from the same authors has reduced the computation problem by analysing several types of fault classifications and relationships. This approach exploits package dependency, fault dependency, fault keywords, fault classifications to seek the relationships between fault causes. These approaches have been evaluated on software bug datasets obtained from different open source bug tracking systems. Sinnamon et al. [6] have applied the binary decision diagram to identify system failure and reliability. Large systems

usually produce thousands of events that consume a large amount of processing time. This diagram associated with if-then-else rules and optimized techniques reduces time consuming problem. The study [7] has proposed an analysis strategy aiming at increasing the likelihood of obtaining a binary decision diagram for any given fault tree while ensuring the associated calculations as efficient as possible. The strategy contains 2 steps: simplifying the fault tree structure and obtaining the associated binary decision diagram. The study also includes quantitative analysis on the set of binary decision diagrams to obtain the probability of top events, the system unconditional failure intensity and the criticality of the basic events. The authors of the study [8] have presented two new tree-based techniques for refining the initial classification of software failures based on their causes. The first technique uses tree-like diagrams to represent the results of hierarchical cluster analysis. The second technique refines an initial failure classification that relies on generating a classification tree to recognize failed executions. This technique uses classification and regression tree for each subject of programs. Zheng et al. [9] has presented a decision tree learning approach based on the C4.5 algorithm to diagnose failures in large Internet sites. The approach records runtime properties of each request and applies automated machine learning and data mining techniques to identify the causes of failures. The approach has been evaluated on application log datasets obtained from the eBay centralized application logging framework.

Classification and regression trees (CART) [3] have been introduced by Breiman et al. and widely been used in data mining. Two main types of decision trees are classification and regression trees. The former tree predicts the outcome that belongs to one of the classes of the input data, e.g., predicting that today's weather is sunny, rainy or cloudy, while the later tree predicts the outcome that can be considered a real number, e.g., predicting that today's temperature is 25.3, 27.5, or 29.7 degree Celsius. Trees used for regression and classification have some similarities and also differences, such as the procedure used to determine where to split. There are several variants of decision tree algorithms. Iterative Dichotomiser 3 (ID3) [10] was developed in 1986 by J. R. Quinlan. This algorithm creates a multi-level tree that seeks a categorical feature for each node using a greedy method. The features yield the largest information gain for categorical targets. Trees are grown to their maximum size and then applied to generalise to unseen data. The algorithm C4.5 [11] is an extension of the ID3 algorithm that converts the trained trees as the output of the ID3 algorithm into sets of if-then rules. Evaluating the accuracy of rules determines the order in which these rules are applied. Instead of finding categorical features, this algorithm uses numerical variables to define a discrete attribute and partitions the continuous attribute values into a discrete set of intervals. Chi-squared automatic interaction detector (CHAID) [12] use multi-level splits to compute classification trees. This algorithm focuses on categorical predictors and targets. It computes a chi-square test between the target variable and each available predictor and then uses the best predictor to partition the sample into segments. It repeats the process with each segment until no significant splits remain. There

are several differences between the CHAID and CART algorithms: (i) CHAID uses the chi-square measure to identify splits, whereas CART uses the Gini or Entropy rule; (ii) CHAID supports multi-level splits for predictors with more than two levels, whereas CART supports binary splits only and identifies the best binary split for complex categorical or continuous predictors; (iii) CHAID does not prune the tree, whereas CART prunes the tree by testing it against an independent (validation) data set or through n-fold cross-validation.

3 CART Approach

The CART approach [3] uses a binary recursive partitioning process to build a decision tree. This process starts with the root node where data are split into two children nodes and each of the children node is in turn split into grandchildren nodes. The process runs recursively until no further splits are possible due to lack of data and the tree reaches a maximal size. The process deals with continuous and nominal features as targets and predictors.

3.1 Entropy Splitting Rule

A decision tree is built top-down from a root node and involves partitioning data into subsets that contain instances with similar values (homogeneous). The CART algorithm uses entropy to calculate the homogeneity of a sample.

$$H(S) = -\sum_{x \in X} P(x) \log P(x) \quad (1)$$

where, S is the current (data) set for which entropy is being calculated. X is a set of classes in S . $P(x)$ is the proportion of the number of elements in class x to the number of elements in set S . When $H(S) = 0$ the set S is perfectly classified.

Information gain $IG(A, S)$ is the measure of the difference in entropy from before to after the set S is split on an attribute A . In other words, how much uncertainty in S was reduced after splitting set S on attribute A .

$$IG(A, S) = H(S) - \sum_{t \in T} P(t) H(t) \quad (2)$$

where, $H(S)$ is entropy of set S . T is the subset created from splitting set S by attribute A . $P(t)$ is the proportion of the number of elements in t to the number of elements in set S . $H(t)$ is entropy of subset t . Information gain can be calculated (instead of entropy) for each remaining attribute. The attribute with the largest information gain is used to split the set S on this iteration.

3.2 Tree Growing Process

The tree growing process uses a set of data features as input. A feature can be ordinal categorical, nominal categorical or continuous. The process chooses the best split among all the possible splits that consist of possible splits of each

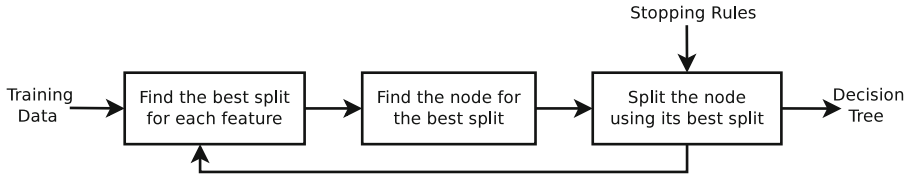


Fig. 1. A process of growing a CART decision tree

feature, resulting in two subsets of data features. Each split depends on the value of only one feature. The process starts with the root node of the tree and repeatedly runs three steps on each node to grow the tree, as shown in Fig. 1.

The first step is to find the best split of each feature. Since feature values can be computed and sorted to examine candidate splits, the best split maximizes the defined splitting criterion. The second step is to find the best split of the node among the best splits found in the first step. The best split also maximizes the defined splitting criterion. The third step is to split the node using its best split found in the second step if the stopping rules are not satisfied. Several stopping rules are used:

- If a node becomes pure; that is, all cases in a node have identical values of the dependent variable, the node will not be split.
- If all cases in a node have identical values for each predictor, the node will not be split.
- If the current tree depth reaches the user-specified maximum tree depth limit value, the tree growing process will stop.
- If the size of a node is less than the user-specified minimum node size value, the node will not be split.
- If the split of a node results in a child node whose node size is less than the user specified minimum child node size value, the node will not be split.

Figure 6 plots a sample CART tree with 4 levels (refer to the end of the paper). The tree grows enormously as the data size increases.

4 Fault Data Analysis

Fault data analysis in this study focuses on using a decision tree to evaluate the severity level of potential fault cases, such as bug reports, log events or trace messages. We have used a bug report dataset for analysis because bug reports are already verified while log events are not verified yet.

4.1 Bug Data

Bug data contains software and hardware bug reports obtained from forums, archives and BTSs. Several tracker sites available on the Internet, such as

Table 1. Popular bug tracking sites (as of November 2014). A plus indicates that the numbers present a lower bound

Tracker site	System	Bugs
bugs.debian.org	Debian BTS	900.000 ⁺
bugs.kde.org	Bugzilla	400.000 ⁺
bugs.eclipse.org	Bugzilla	400.000 ⁺
bugs.gentoo.org	Bugzilla	350.000 ⁺
bugzilla.mozilla.org	Bugzilla	800.000 ⁺
bugzilla.redhat.com	Bugzilla	900.000 ⁺
qa.netbeans.org	Bugzilla	250.000 ⁺
bugs.launchpad.net	Launchpad	1.200.000 ⁺

Table 2. List of important features

Feature	Description	Data types
Status	The open, fixed or closed status of the bug	Enumerate
Component	The component contains the bug	Enumerate
Software	The software contains the bug	Enumerate
Platform	The platform where the bug occurs	Enumerate
Keyword	The list of keywords that describe the bug	Text
Relation	The list of bugs related to the bug	Numeric
Category	The category of the bug	Enumerate

Bugzilla [13], Launchpad [14], Mantis [15], Debian [16] provide web interfaces to their bug data. Tracker sites differ from data inclusion and presentation, but share several similar administration and description fields. While the administration fields are represented as field-value pairs, such as severity, status, platform, content, component and keyword, the problem description field details the problem and follow-up discussions represented as textual attachments. We have used a web crawler to get as much access to bug data as ordinary users. The crawler retrieves the HTML pages of bug reports, then few parsers extract the content of bug reports and save the content to a database following a unified bug schema [17]. Table 1 reports popular BTSs and numbers of downloadable bug reports for tracker sites.

A bug report contains several features shown in the unified bug schema. Some features cause less impact on determining the severity of the bug report, such as owner, created time, updated time, etc. Our approach therefore focuses on the features as shown in Table 2. Note that each bug report contains the severity feature with a value. It is necessary to ignore this feature when building the tree to avoid some side effect. The keyword feature that contains the description and discussion of the bug requires further data processing.

4.2 Data Processing

Processing features improves the quality of the training datasets and thus enhance the performance of the decision tree. A bug report contains a textual part of the problem description and some discussions that hide distinct keywords or groups of keywords. We have applied the term frequency–inverse document frequency (tf×idf) method to reveal these keywords for the keyword feature. This method measures the significance of keywords to bug reports in a bug dataset by the occurrence frequency of the keywords in a bug report over the total number of the keywords of the bug report (term frequency) and the occurrence frequency of the keywords in other bug reports over the total number of bug reports (inverse document frequency). A distinct group of keywords contains related keywords with high significance. As a consequence, the keyword feature includes a set of keywords and groups that best describe the bug report. However, since bug reports are obtained from various BTSs, their descriptions and discussions contain redundant words, nonsense words or even meaningless words, such as: memory address, debug information, system path, article, etc. Algorithm 1 filters out these words from the bug dataset. We have implemented this algorithm in Python programming language.

Algorithm 1. Filtering keywords for a bug dataset

Input : Raw keyword set

Output: Filtered keyword set

- 1 Load raw keyword set;
 - 2 Remove duplicated words and redundant words by using stop-word set;
 - 3 Remove meaningless words by using regular expression;
 - 4 Remove memory addresses by filtering special characters;
 - 5 Process tf×idf on the whole keyword set;
 - 6 **return** Filtered keyword set;
-

The first step is to load the bug dataset focusing on the keyword feature. The next three steps are to filter useless keywords. The stop-word set is the set of popular keywords that usually appear in textual description such as a, an, the, of, etc. The regular expression contains characters [0–9], [a-f] and [A-F], while the special characters contains `_`, `-`, `\`. The final step is to apply the tf×idf method on the whole keyword set and remove trivial keywords, i.e., keywords with low tf×idf values.

4.3 Tree Construction

The previous section explains using Entropy splitting rule to grow a decision tree. We present in this section using Scikit Learn library [18] to construct decision trees for bug datasets. Scikit Learn is an open source machine learning library for Python programming language and provides several classification, regression and

clustering algorithms. It is designed to interoperate with Python numerical and scientific libraries such as NumPy [19] and SciPy [20]. The CART algorithm is one of the main classification algorithms supported by Scikit Learn. Algorithm 2 presents main steps to construct decision trees using the Scikit Learn library:

Algorithm 2. Constructing a decision tree for a bug dataset

Input : Processed bug dataset

Output: Decision tree

- 1 Load the dataset into pandas data-frame and drop the platform feature;
 - 2 Factorize the features;
 - 3 Load sample data and class label;
 - 4 Split the dataset into the training set and testing set;
 - 5 Fit the training set into decision tree classifier;
 - 6 Construct the tree using entropy criterion;
-

The first step is to load the dataset into pandas data-frame that is a special tabular data structure to prepare data for the CART algorithm. It is also important to drop the platform feature in the data-frame because the dataset is already grouped by this feature. Since the CART algorithm cannot deal with non-numerical values, while the feature values in the bug dataset are non-numeric, i.e. enumerate or text, all the feature values need to be factorized into numerical values in the second step. The pandas library supports for converting non-numerical values to numerical values. Each distinct value is replaced by a unique integer, e.g. the severity feature contains 4 values: feature, minor, normal and critical corresponding to 0, 1, 2, 3 after factorization. The next step is to separate the data-frame into 2 parts. The first part is the sample data that contains the numerical values of all features, while the second part is the class label that marks the numerical classes for each particular bug. The most important step in this algorithm is to partition the sample data and class label into the training set and testing set. The training set is used for training the decision tree, while the testing set is used for evaluating the decision tree. The percentages of the training and testing sets are 75% and 25% respectively. Finally, the decision tree is trained by a method supported by Scikit Learn library. The input of this method is the training set found in the previous step. Figure 7 plots a part of a decision tree for the Linux platform dataset (refer to the end of the paper).

Since the decision tree contains multiple levels, we only present the first 4 levels. The leaf nodes contains the following values:

1. The first component counts samples that have the severity of feature
2. The second component counts samples that have the severity of critical
3. The third component counts samples that have the severity of minor
4. The fourth component counts samples that have the severity of normal

5 Evaluation

We have used a dataset of 130.000 bug reports for experiments. A large dataset usually results in large decision tree that possibly causes performance problem

due to the complexity and memory consumption of the tree. The authors of the study [21] have already proposed an approach to construct decision trees from very large datasets. The approach builds a set of decision trees based on tractable size training datasets which are subsets of the original dataset. The result of the study also reveals the over-fitting issue of a large decision tree. We have separated bug reports into 4 smaller datasets following the platform feature: 50.000 bug reports occurring on all platforms (All platform), 50.000 bug reports occurring on Windows platform (Win platform), 15.000 bug report occurring on Linux platform (Linux platform) and 15.000 bug report occurring on Macintosh platform (Mac platform). We have built decision trees on parallel and performed all experiments on a workstation with Intel i5-2450M CPU 2.5 GHZ, 4 GB of RAM and Ubuntu 12.04.

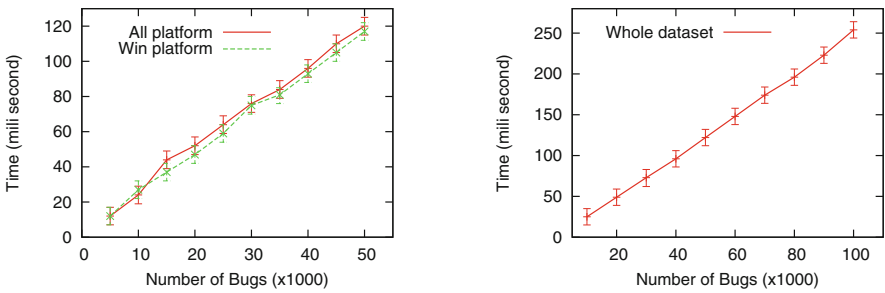


Fig. 2. Time consumption for constructing decision trees over various datasets

The first experiment measures time consumption for constructing decision trees over various datasets. Time consumption linearly increases as the size of datasets increases, as shown in Fig. 2. It takes approximately 120 ms or 250 ms to build a decision tree for 50.000 or 100.000 bug reports, respectively. Note that time consumption depends on numbers of events and features. Bug reports in the Win platform dataset contain less one feature than bug reports in the whole dataset, i.e., the platform feature, thus time consumption for both datasets is slightly different. Since processing large event log files that usually contain millions of events consumes much time, reducing processing time is necessary.

The datasets contain thousands of bug reports that possibly miss several features, it is necessary to apply the median imputation method for datasets to improve performance. Dealing with missing values is one of the most common issues in data training process. It occurs when data values are unavailable for observations due to the lack of responses: data is provided for neither several features nor a whole case. Missing values are sometimes caused by researchers, e.g., data collection is done improperly or mistakes are made by input data.

Using training datasets with missing values can affect performance in classification. Several prevailing methods deal with this issue. Case deletion method discards cases with missing values for at least one feature. A variant of this methods only eliminates cases with a high level of missing values while determining

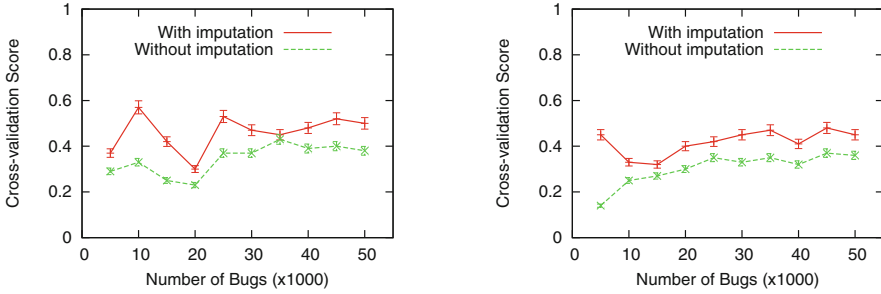


Fig. 3. Cross-validation comparison for the All platform (left) for the Win platform (right) with and without imputation

the extent of features for cases with a low level of missing values. Mean imputation method replaces missing values for features by the mean of all known values of the features in the class to which the case with missing values belongs. Similar to the mean imputation method, the median imputation method replaces missing values for features by the median of all known values of the features in the class to which the case with missing values belongs. Using median avoids the presence of outliers and also assures the robustness of the method. This method is suitable for datasets that the distribution of the values of a certain feature is skewed. Modified K-nearest neighbor method determines missing values for a case by considering a certain number of the most similar cases. The similarity of two cases is measured by a distance function.

The second experiment fills missing values for bug reports using the mean imputation method and then compares cross-validation scores for both datasets with and without imputation. Figure 3 on the left side reports low and unstable scores for the All platform dataset, especially the score reduces to 0.3 approximately for both datasets of 20,000 bugs. The All platform dataset with imputation obtains the average cross-validation score of 0.5 that improves a reasonable number of missing values from the All platform dataset without imputation. The Win platform dataset performs worse than the All platform dataset as shown in Fig. 3 on the right side. The Win platform dataset with imputation obtains the average cross-validation score of 0.4 that improves a lower number of missing values compared with the All platform dataset. Note that the number of missing values increases as the size of datasets increases, thus using the imputation technique can improve the accuracy score of prediction.

The third experiment focuses on the accuracy of the decision tree. The idea is to divide the original dataset with imputation into the training and testing datasets. While the training dataset is used to build the decision tree, the testing dataset is used to evaluate the accuracy of the decision tree. The extreme case of cross-validation, namely leave-one-out cross-validation, has been used for this experiment. We have used the decision tree to predict the severity level of a bug report and built a list of the predicted severity levels for bug reports in the testing dataset. This list is then compared with the list of the correct severity

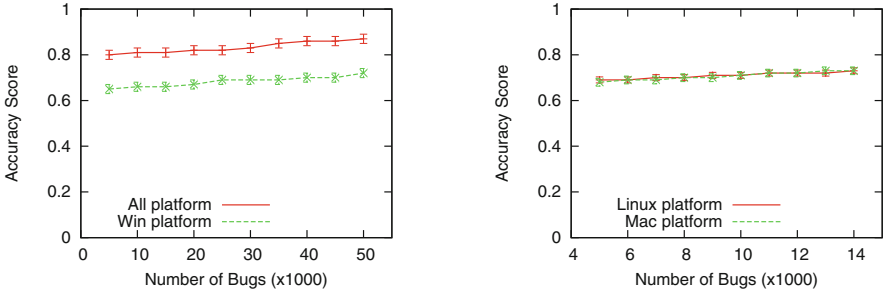


Fig. 4. Accuracy comparison between the All and Win platforms (left) and between the Linux and Mac platforms (right)

levels of the testing dataset. Accuracy score is calculated based on the number of matching severity levels in 2 lists.

Figure 4 on the left side reports high and stable accuracy scores for the All platform dataset with the average score of 0.82 approximately. The All platform dataset also outperforms the Win platform dataset that obtains the average score of 0.68 approximately. We observed that bug reports for all platforms tend to be common problems that can be easily reproduced, determining severity levels for these bug reports is rather straightforward and precise. On the other hand, bug reports for certain platforms are sometimes very specific and difficult to determine a severity level properly. The accuracy scores for the Linux and Mac platform datasets share the same explanation with the Win platform dataset. Figure 4 on the right side presents the similar accuracy scores of both datasets with the average score of 0.71 approximately. However, the Win platform dataset is larger than the Mac and Linux platform datasets that can cause an impact on accuracy scores as the size of datasets increases. In addition, average time to train the decision tree of 50.000 bug reports is considerably fast.

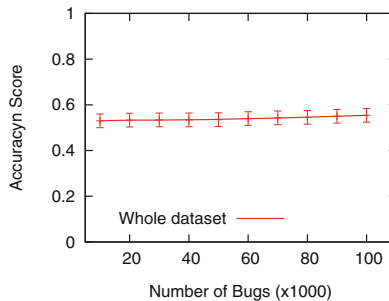


Fig. 5. Accuracy score for the whole dataset



Fig. 6. A sample CART tree

Figure 5 also plots the accuracy score of the whole dataset. Since bug reports of this dataset contain the platform feature, the decision tree is therefore larger and more complex than the above trees. The accuracy score is lower and more stable than the above accuracy scores on average. Larger and more complex decision trees possibly yield low efficiency due to the low quality of large datasets, e.g., a dataset with several missing values.

6 Conclusions

We have proposed an approach of using the CART decision tree for fault data analytics that can be applied to event monitoring and fault detection in communication networks and distributed systems. The event log datasets are so huge that system administrators and even supporting tools may ignore potentially critical events. This decision tree is characterized by the capability of learning from the training datasets and then determining the severity level of log events from the testing datasets. We have used bug report datasets for experiments. Bug reports obtained from BTSs are to some extent similar to log events with a severity level. Evaluating the approach focuses on the performance and efficiency of the decision tree. We have computed the time consumption of building the tree, the precision of classification and the imputation of missing values. The experimental results reveal that the accuracy scores for different trees are above 70 %, especially 80 % for the all platform dataset. Applying methods to deal with missing values in the training datasets improves efficiency. Moreover, trees with tractable size training datasets consume less processing time and possibly yield high efficiency. Future work focuses on two issues. The first issue uses more features in bug reports or log events, especially exploiting distinct keywords from textual description. The second issue evaluates the efficiency of larger trees built by larger training datasets.

Acknowledgements. This research activity is funded by Vietnam National University in Ho Chi Minh City (VNU-HCM) under the grant number C2015-28-02

References

1. Buyya, R., Yeo, C.S., Venugopal, S., Broberg, J., Brandic, I.: Cloud computing and emerging it platforms: vision, hype, and reality for delivering computing as the 5th utility. *Future Gener. Comput. Syst.* **25**(6), 599–616 (2009)
2. Armbrust, M., Fox, A., Griffith, R., Joseph, A.D., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I., Zaharia, M.: A view of cloud computing. *ACM Commun.* **53**(4), 50–58 (2010)
3. Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J.: *Classification and Regression Trees*. Chapman & Hall/CRC, New York (1984)
4. Tran, H.M., Schönwälder, J.: Fault representation in case-based reasoning. In: Clemm, A., Granville, L.Z., Stadler, R. (eds.) *DSOM 2007*. LNCS, vol. 4785, pp. 50–61. Springer, Heidelberg (2007)

5. Tran, H.M., Le, S.T., Ha, S.V.U., Huynh, T.K.: Software bug ontology supporting bug search on peer-to-peer networks. In: Proceeding 6th International KES Conference on Agents and Multi-agent Systems Technologies and Applications (AMSTA 2013). IOS Press (2013)
6. Sinnamon, R.M., Andrews, J.D.: Fault tree analysis and binary decision diagrams. In: Proceeding in Reliability and Maintainability Annual Symposium, pp. 215–222 (1996)
7. Reay, K.A., Andrews, J.D.: A fault tree analysis strategy using binary decision diagrams. *Reliab. Eng. Syst. Saf.* **78**(1), 45–56 (2002)
8. Francis, P., Leon, D., Minch, M., Podgurski, A.: Tree-based methods for classifying software failures. In: Proceedings of 15th International Symposium on Software Reliability Engineering (ISSRE 2004), pp. 451–462. IEEE, Washington (2004)
9. Zheng, A.X., Lloyd, J., Brewer, E.: Failure diagnosis using decision trees. In: Proceeding of 1st International Conference on Autonomic Computing (ICAC 2004), pp. 36–43. IEEE Computer Society, Washington (2004)
10. Quinlan, J.R.: Induction of decision trees. *Mach. Learn.* **1**(1), 81–106 (1986)
11. Quinlan, J.R.: C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, San Francisco (1993)
12. Kass, G.V.: An exploratory technique for investigating large quantities of categorical data. *Appl. Stat.* **29**(2), 119–127 (1980)
13. Mozilla Bug Tracking System. <https://bugzilla.mozilla.org/>. Accessed Jan 2015
14. Launchpad Bugs. <https://bugs.launchpad.net/>. Accessed Jan 2015
15. Mantis Bug Tracker. <https://www.mantisbt.org/>. Accessed Jan 2015
16. Debian Bug Tracking System. <https://www.debian.org/Bugs/>. Accessed Jan 2015
17. Tran, H.M., Lange, C., Chulkov, G., Schönwälder, J., Kohlhase, M.: Applying semantic techniques to search and analyze bug tracking data. *J. Netw. Syst. Manag.* **17**(3), 285–308 (2009)
18. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: machine learning in python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011)
19. Oliphant, T.: A guide to NumPy, vol. 1. Trelgol Publishing, USA (2006)
20. Silva, F.B.: Learning SciPy for Numerical and Scientific Computing. Packt Publishing, Birmingham (2013)
21. Hall, L.O., Chawla, N., Bowyer, K.W.: Decision tree learning on very large data sets. In: Proceedings of IEEE International Conference on Systems, Man and Cybernetics, vol. 3, pp. 2579–2584. IEEE (1998)

Evaluation of Reliability and Security of the Address Resolution Protocol

Elvia León^{1(✉)}, Brayan S. Reyes Daza²,
and Octavio J. Salcedo Parra^{1,2}

¹ Universidad Nacional de Colombia, Bogotá D.C., Colombia
ejleonmu@unal.edu.co, osalcedo@udistrital.edu.co

² Universidad Distrital Francisco José de Caldas, Bogotá D.C., Colombia
bsreyesd@correo.udistrital.edu.co

Abstract. This article shows the procedure to execute an ARP poisoning in order to stand out the insecurity that the Protocol has, and to compare it against other alternatives, to show the safety of each of these. Where it is concluded that ES-ARP and S-ARP are good choices to improve the safety of the ARP protocol, although is not 100 % secure, since if they send the answer and then the poisoned ARP reply is sent before the actual one is received, and set on the cache memory, the victim stores the wrong response in the cache and discards the actual one. When the first ARP request is sent, the victim and the attacker receive the message. Who comes first will get the ARP cache of the victim.

Keywords: MAC address · ARP · ES-ARP · S-ARP

1 Introduction

Address Resolution Protocol is a complicated protocol. “Many implementations do not interpret the Protocol specification, and others supply wrong links since they eliminate the cache timeout in an attempt to improve efficiency”. We can say that the ARP protocol works mainly is three parts, the first that returns a link that is used by the network interface to encapsulate and transmit the package. Another module that manages the ARP packets that arrive from the network and update the ARP cache by adding new links. And finally a manager who implements the policy of replacing cache, which examines entries in the cache, and removes them when they reach a certain time.

When an ARP reply is requested, this can be changed (MAC address), by a false one, and thus attacking the system, this is called ARP poisoning. Due to this poisoning, alternatives have been sought to make the protocol secure, one of them is S-ARP, that is based on an extension of the ARP protocol and a set of features that allow a verification of authenticity and integrity of the contents of the ARP replies are introduced using asymmetric cryptography. And there also ES-ARP guided mainly by the greater problem of ARP that is the fact that this is totally gullible, since it doesn't difference between the received messages and it confides blindly into what it has received, since it is a stateless protocol and does not carry any information about the requests sent or the responses received This “loop” is used by attackers to submit falsified answers so they are accepted by the ARP and end up poisoning the ARP cache. S ARP has been

implemented for some time, but did not succeed due to the fact that when it sends two packets of data confirmation messages, makes this authentication method slower than normal, which decreases efficiency. Instead the implementation of ES-ARP only has been made to be tested and it has not come out to the public, according to their results, it does not have any inconvenience for its implementation.

2 Address Resolution Protocol

Address Resolution Protocol (ARP) is a stateless protocol, i.e., a response can be processed although the request was never received. When a response is received, the corresponding entry in the cache is updated with the logical IP address, and the physical MAC address in the answer. An ARP is a message sent by a host requesting the MAC addresses its own IP address, it is sent from one machine to another with the same or different network. ARP resides in the network layer of the TCP/IP suite, where a host is identified by its 32-bit IP address, and the MAC address that follows a scheme of 49 bits.

When the network layer receives a message from the upper layers, it checks the IP address of the target machine. If the target machine is on the same local network as the of source machine, the message can be sent directly to the target machine, but on the other hand if you are not in a local network the message has to be aimed through a router. To send the message directly to the target machine, the network layer needs to know the MAC address of the target machine. ARP dynamically allocates the 32-bit IP address of a machine to its 48 bits MAC address in a temporary memory location called the ARP cache space.

There are two types of ARP messages that may be sent by the ARP protocol. One is ARP request and the other is ARP reply.

- ARP request: when the ARP request is done through a host, the IP address, MAC address, ARP message type and the destination IP address are framed. This request is diffused to all hosts on the same LAN as the sending host, the destination MAC address field is left blank for the host with the IP address of destination to fill it out.
- ARP response: when a host receives the ARP request containing the IP address as the destination IP address, which is filled with MAC address and the field of operation is set on the operation code of the ARP response. This message is sent directly to the requesting machine. When the ARP reply is received by the requesting machine, it updates its ARP cache with the requested MAC address.

When creating an ARP response, an attacker can easily change the Association maintained in the ARP cache of host, that is, can change the MAC address by a false one (false response) sending IP encapsulated messages with this false address. In this way the attacker can receive all frames originally directed to another host. Once the host is poisoned, it will send all traffic to the attacker host s can read them, and if it decides to forward them back the attacked host they will not realize that they are being attacked. This is an attack MITM (man in the middle). Another attack on this network is DoS (denial of service) is when the attacker does not forward messages after reading them, to the target machine, and this is called a denial of service attack.

2.1 ES-ARP

ES-ARP is guided mainly by the greater problem of ARP that is the fact that this is totally gullible, since it doesn't difference between the received messages and it confides blindly into what it has received, since it is a stateless protocol and does not carry any information about the requests sent or the responses received This "loop" is used by attackers to submit falsified answers so they are accepted by the ARP and end up poisoning the ARP cache.

ES-ARP implements a method in such a way that the "the ARP request and ARP response is transmitted, and is storing the application plot information in the ARP cache. In this Protocol, all hosts except the source host will store entries in the ARP cache" (Md. Atullah, N. Chauhan).

In Table 1 we can observe the steps of request and response of ARP and ES-ARP, which shows us some differences that we have in ES-ARP in order to improve the security of the data that is sent in ARP, such as making all machines on the local network to receive the ARP request transmitted and immediately update their ARP cache with the MAC address of the source device, to be checked in its ARP cache and make sure if the target host input is there or not. Which makes that once the MAC address is stored it does not allow that they falsified it thereby preventing them to poison the ARP, and in such way achieving to be safer, since only the source host will accept the response, otherwise it will be simply discard the ARP response plot.

This procedure updates the ARP cache twice, it means the first time that the ARP request is sent in which the IP will be stored and the MAC of the source host, and the second time when the ARP answer is issued which means that the IP and the MAC of the target host are stored.

Table 1. Procedures for request and response of ARP Y ES-ARP.

ARP	ES-ARP
1. An A machine wants to send a packet to D, but A only knows the IP address of D	1. An A machine wants to send a packet to D, but A only knows the IP address of D
2. Machine A broadcast ARP request with the IP address of D	2. Machine A broadcast ARP request with the IP address of D
3. All machines on the local network receive the ARP request that is issued	3. All machines on the local network receives the ARP request that has been sent and update their ARP cache with the MAC of A
4. D machine responds with its MAC address ARP unicast response and updates its ARP cache with the MAC of A	4. D machine responds with its MAC address through ARP response
5. Machine A adds the MAC address of D to its ARP cache	5. All machines add the MAC address of D to its ARP cache
6. A machine can now offer packages directly to D	6. A machine can now offer packages directly to D

2.2 S-ARP

S ARP is based on an extension of the ARP protocol and introduces a set of features that allow a verification of authenticity and integrity of the contents of the ARP replies, using asymmetric cryptography. In addition also follows the same specifications of ARP, to make it compatible with this and it only inserts a header end of standard protocol messages to carry the authentication information, which means that S-ARP does not accept messages not authenticated (unless you are on a list of known hosts).

S-ARP takes in the ARP response a header S-ARP and ARP requests do not change. S-ARP header contains the digital signature of the sender, a mark of time, the type and the length of the message, the message authenticates looking for the IP address of the sender and its corresponding public key in its ring (since each host maintains a ring of public keys and corresponding IP addresses previously requested by the AKD (Authorized dealer key)), if the input uses the contents to check the signature, otherwise, sends a request to the AKD for certification.

A request to the AKD is also sent when the key on the local ring does not check the signature, since it may no longer be valid. In this case, the packet in queue is a “list of pending answers”. The AKD sends a response signed with the public key requested and current time stamp. When you receive the response of the AKD, host synchronizes the local with the time stamp clock, if necessary, it stores the public key in its ring, and verifies the signature... In case the old key is no longer valid, if the new key received from the AKD is the same as the one in the cache, the response is considered invalid and is dropped. If the key has changed in fact, the host refreshes its cache and verifies the signature with the new key.


3 Methodology

What implemented an ARP poisoning, more precisely a “man in the middle attack” or man in the middle MITM, the attacker computer has a Realtek PCIe GBE network card 802 controller family. 11B, with processing speed of 2.1 GHz, and Windows 8.1 operating system.

We used a network sniffer called Cain and Abel this program takes advantage of some insecurity presented in the ARP protocol standards and its method of authentication; its main purpose is the simplified recovery of passwords and credentials from various sources, but also has some not standards utilities for Microsoft Windows users. Features such as ABRIL (Arp poison routing) which enables sniffing in LANs (cheating the switch tables) (Fig. 1).

4 Implementation

First we need to start the application (Figs. 2, 3, 4, 5 and 6).

Now we will intercept the traffic generated by the victim, by choosing within Cain sniffer module, the APR option and then by clicking on the symbol  and would deploy a window (Fig. 7).

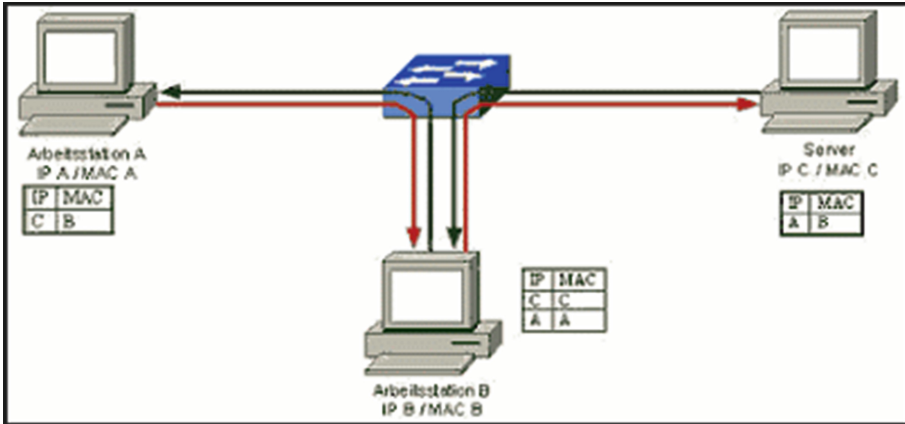


Fig. 1. Attack MITM where the victim access to the router, the router to the attacker, then goes to the router again and finally to the server and receives a response from the server to the router, then to the attacker, from the attacker to the router and then to the victim.

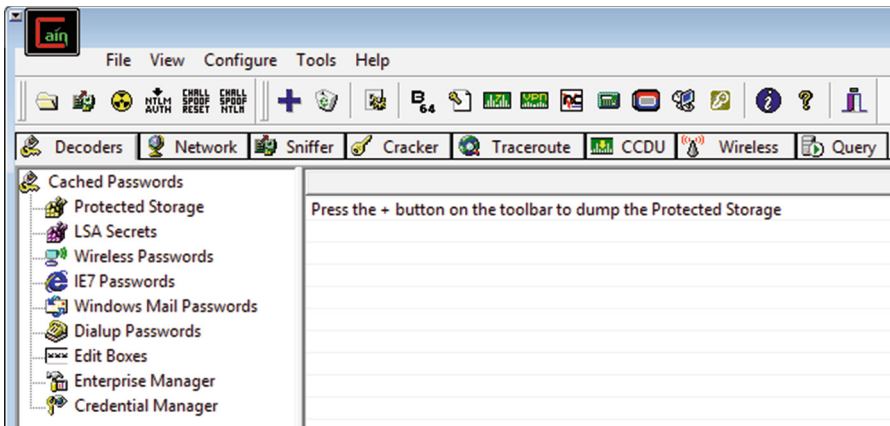


Fig. 2. We start by booting Windows 8 to Cain and Abel.

Once added the victim to the list, we will click on the button 🚫 to begin to redirect the traffic. This will cause that the offensive computer will temporarily copy the IP of the gateway and get packets destined to it by the customer, without interrupting the connection between the two of them, so that the customer does not notice any change (Fig. 8).

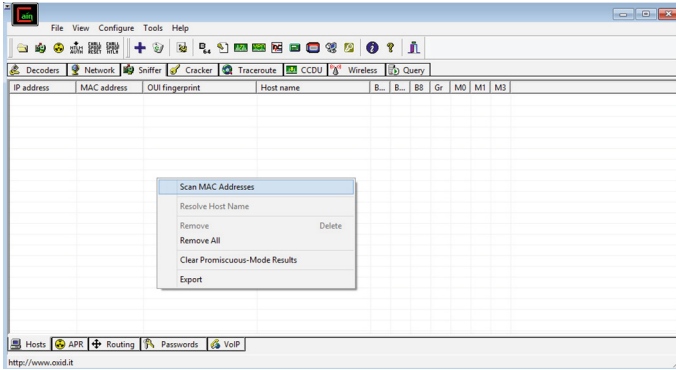


Fig. 3. In the toolbar, select Star Stop Sniffer, and then click on host, then the right click and select Scan MAC Addresses.

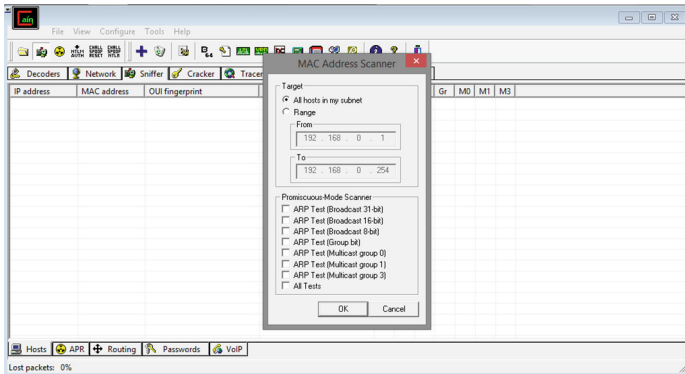


Fig. 4. It will show us a window and we will mark the all hosts option in my subnet and give ok.

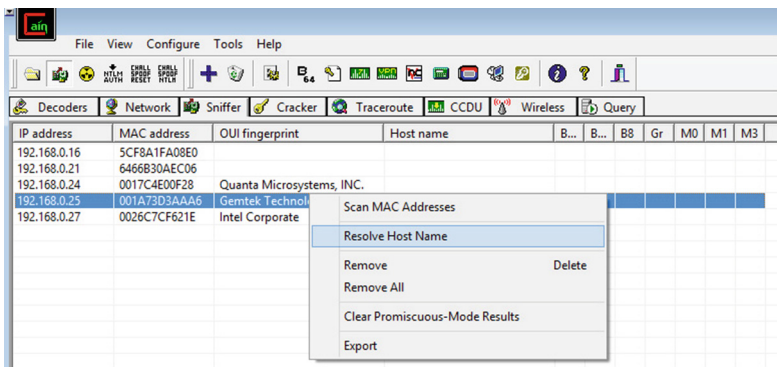


Fig. 5. After giving OK, it would deploy a list with the computers connected in the subnet.

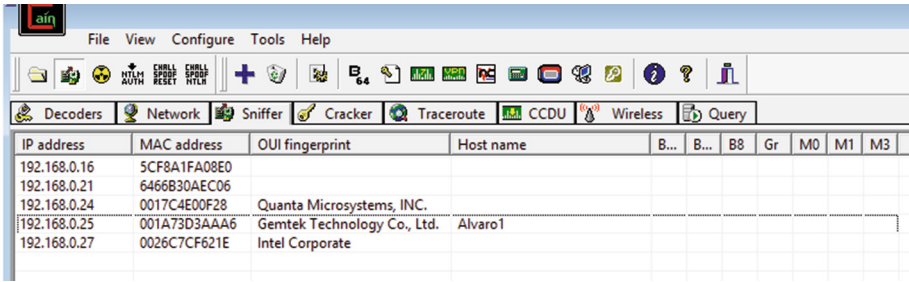


Fig. 6. We can select the team that we want to attack, also we can see the host name, by selecting the computer and giving it right click and Resolve Host Name, and the name will be displayed.

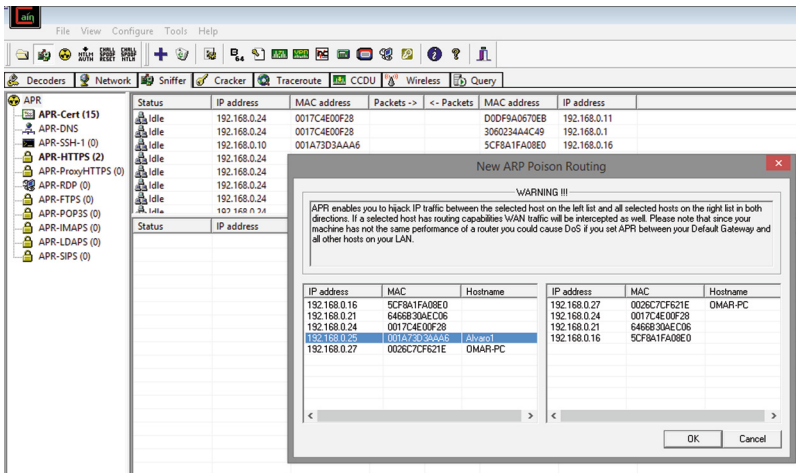


Fig. 7. Shows on the left side the IP of the victim and in the right side, the direction that we want to intervene or the gateway.

5 Security Against Address Resolution Protocol Attacks

There are different ways to make the ARP protocol safer, in addition to the two mentioned above, it can also be through programs or the use of our computer commands console. We can use a tool called Arpoon (Arp handler inspectiON) is a tool that allows you to manipulate certain aspects of the ARP Protocol. One of his outstanding qualities is to make the ARP protocol safer and it Implements two techniques of defense against ARP poisoning attacks (ARP spoofing): SARPI “Static Arp Inspection”: Static ARP inspection: networks without DHCP. It uses a static list of entries and no modifications are allowed. And DARPI “Dynamic Arp Inspection”: Dynamic ARP inspection: networks with DHCP. It controls incoming and outgoing ARP requests, saves the outgoing and sets a timeout for the incoming response.

Status	IP address	MAC address	Packets ->	< - Packets	MAC address	IP address
APR-Cert (23)						
APR-DNS						
APR-SM+ (0)						
APR-HTTPS (7)						
APR-ProxyHTTPS (0)						
APR-POP (0)						
APR-FTPS (0)						
APR-POP3S (0)						
APR-IMAPS (0)						
APR-LDAPS (0)						
APR-SIPS (0)						
Full-routing	192.168.0.24	0017C4E00F28	11	11	3060234A4C49	190.85.253.97
Half-routing	192.168.0.24	0017C4E00F28	1	0	3060234A4C49	74.125.137.189
Half-routing	192.168.0.24	0017C4E00F28	2	0	3060234A4C49	173.194.125.22
Full-routing	192.168.0.24	0017C4E00F28	22	5	3060234A4C49	165.254.42.75
Full-routing	192.168.0.24	0017C4E00F28	8	2	3060234A4C49	179.60.192.181
Full-routing	192.168.0.24	0017C4E00F28	175	21	3060234A4C49	31.13.69.160
Half-routing	192.168.0.24	0017C4E00F28	4	0	3060234A4C49	77.234.42.60
Full-routing	192.168.0.24	0017C4E00F28	44	11	3060234A4C49	181.48.0.229
Full-routing	192.168.0.24	0017C4E00F28	5	2	3060234A4C49	190.85.253.17
Half-routing	192.168.0.24	0017C4E00F28	6	0	3060234A4C49	179.60.192.197
Half-routing	192.168.0.24	0017C4E00F28	7	0	3060234A4C49	31.13.69.80
Half-routing	192.168.0.24	0017C4E00F28	17	0	3060234A4C49	192.204.3.73
Full-routing	192.168.0.24	0017C4E00F28	98	14	3060234A4C49	209.48.37.49
Full-routing	192.168.0.24	0017C4E00F28	29	1	3060234A4C49	72.246.65.124
Full-routing	192.168.0.24	0017C4E00F28	8	3	3060234A4C49	190.157.8.33
Full-routing	192.168.0.24	0017C4E00F28	236	35	3060234A4C49	179.60.192.229
Half-routing	192.168.0.24	0017C4E00F28	58	0	3060234A4C49	65.55.53.190
Full-routing	192.168.0.24	0017C4E00F28	180	42	3060234A4C49	190.93.253.80
Full-routing	192.168.0.24	0017C4E00F28	103	5	3060234A4C49	23.73.180.17
Half-routing	192.168.0.24	0017C4E00F28	9	0	3060234A4C49	23.73.180.19
Full-routing	192.168.0.24	0017C4E00F28	50	5	3060234A4C49	96.6.113.146
Full-routing	192.168.0.24	0017C4E00F28	20	1	3060234A4C49	179.60.192.149
Full-routing	192.168.0.24	0017C4E00F28	39	15	3060234A4C49	165.254.40.130

Fig. 8. Once made throughout the procedure, we will be able to extract the data from the victim machine, we can capture passwords of unsafe websites or password protocols FTP, VNC etc.

In addition to detect and block more complex derived attacks such as DHCP Spoofing, DNS Spoofing, WEB Spoofing and Session Hijacking SSL/TLS Hijacking. Is designed to run as a Daemon, and is currently adapted for GNU/Linux, Mac OS X, FreeBSD, NetBSD and OpenBSD systems (GLOBALIP S. A. C, 2011).

6 Debate

ES-ARP works with a method in such a way that the “the ARP request and ARP response is transmitted, and it is storing the request plot information in the ARP cache. What makes that the performance does not decrease. S ARP works with asymmetric cryptography which means that it uses a pair of keys for sending messages (a public key and private of the same host), what makes that the execution time is dominated by the verification of the signature and signature generation, this time of verification depends on the length of the key. The creation of the firm takes a long time due to the exponential calculate but can improve calculating everything separately but still does not significantly improve the performance.

The defense of ES-ARP against poison is by storing the information in the plot of ARP request, which reduces the possibilities of the different types of attacks. Retransmission of the response of the ARP plot provides security against the ARP cache poisoning, as if any attacker would send a false ARP response, then this reply is also received by the target host, whose IP address is used to map the MAC address of the attacker. So this host detects that this ARP reply is false by the attacker. Instead S-ARP uses static entries in the ARP cache, these cannot be updated, and only they can

be changed manually by the system administrator, it is not viable to networks with hundreds of hosts because the inputs must be entered manually on each host. Another suggested S-ARP security option is the safety of port which is a feature shown in many modern switches that allows the switch to recognize only a MAC address on a physical port, still is not effective protection against ARP poisoning, since if the attacker does not falsified its own MAC address, it can poison the cache of the two victims without letting the switch to interfere in the process.

Besides S - ARP has been implemented in a Linux operating system, it is composed of two parts: 1) a patch for the core that eliminates the ARP package from the list of incoming package through the dev remove function. In this way the core will not have to analyze all ARP's packets and release it. The patch does not affect the way in which the core tries to resolve Ethernet addresses, since it continues sending requests normally, only that it will not process the answers. The daemon can act as AKD or as a generic host depending on the parameter of the command line that is passed to the Protocol at the time of the launch. It is also responsible for communications with the AKD for the key management.

7 Conclusions and Recommendations

The two solutions have a problem, if the poisoned ARP reply is sent before the actual response and gets stored in the cache memory, the victim stores the wrong response in the cache memory and discards the actual response. When the first ARP request is sent, the victim and the attacker receive the message. Who comes first will get the ARP cache of the victim.. In addition, the attacker could also impersonate an ICMP echo request message and be sent immediately with a false ARP response. When the victim receives the ICMP echo request, performs an ARP request, but the false response is already in the tail of the packet received, by which is accepted. If an antidote is installed, the host can override the MAC address of the sender and force a series to prohibit another host.

We must know for which type of networks we will adapt the solution since at least with ARP-S it would be tedious to implement it in a network with hundreds of hosts, since entries should be introduced manually on each host, and instead on ES-ARP this feature is not adopted and is similar to ARP. ES-ARP has best performance which makes it more efficient.

References

1. Atullah, Md., Chauhan, N.: ES-ARP: an efficient and secure address resolution protocol. In: SCEECS (2012)
2. Bruschi, D., Ornañu, A., Rosti E.: S-ARP: a secure address resolution protocol. In: ACSAC (2001)
3. Comer, D., Stevens D.: Interconectividad de redes con TCP/IP, vol 11 (2012)
4. Gutiérrez, F.: Laboratorio virtualizado de seguridad informática con Kali Linux (2013)
5. Montoro, M.: Cain y Abel. <http://www.oxid.it/cain.html> (2014)

6. GLOBALIP S.A.C.: “ArpOn” - Un buen aliado contra los ataques AR. <http://globalip.blogspot.com> (2011)
7. Suarez, R.: Seguridad y alta disponibilidad – Manual cain y Abel (sniffer). <http://es.slideshare.net/TotusMuertos/manual-cain-abel-sniffer-en-windows> (2010)
8. Martin, M.: Criterio y funcionamiento de un sniffer cain-Abel. <http://es.slideshare.net/gajul1219/criterio-y-funcionamiento-de-un-sniffer-cain-abel-wwwdragon-jarus> (2015)

Crowdsourcing and Social Network Data Analytics

Establishing a Decision Tool for Business Process Crowdsourcing

Nguyen Hoang Thuan^{1,2(✉)}, Pedro Antunes¹, David Johnstone¹,
and Nguyen Huynh Anh Duy²

¹ School of Information Management, Victoria University of Wellington,
PO Box 600, Wellington, New Zealand

{Thuan. Nguyen, Pedro. Antunes,
David. Johnstone}@vuw.ac.nz

² Can Tho University of Technology,
256 Nguyen van Cu Street, Can Tho, Vietnam
{nhthuan, nhaduy}@ctuet.edu.vn

Abstract. The integration of crowdsourcing in organisations fosters new managerial and business capabilities, especially regarding flexibility and agility of external human resources. However, a crowdsourcing project involves considering multiple contextual factors and choices and dealing with the novelty of the strategy, which makes managerial decisions difficult. This research addresses the problem by proposing a tool supporting business decision-makers in the establishment of crowdsourcing projects. The proposed tool is based on an extensive review of prior research in crowdsourcing and an ontology that standardises the fundamental crowdsourcing concepts, processes, dependencies, constraints, and managerial decisions. In particular, we discuss the architecture of the proposed tool and present two prototypes, one supporting what-if analysis and the other supporting detailed establishment of crowdsourcing processes.

Keywords: Business process crowdsourcing · Crowdsourcing · Decision support system · Design science · Ontology

1 Introduction

Crowdsourcing is becoming a viable, popular business strategy for organisations, which can harness human power, wisdom, information, and ideas from the external crowd in a flexible way and a short period of deployment time [1, 2]. This popularity can be demonstrated by the increasing number of organisations adopting the crowdsourcing strategy and revenues brought by the crowdsourcing market. The list of organisations that successfully adopted crowdsourcing is long, including big companies like iStockPhoto, Amazon, Threadless, Colgate-Palmolive, Unilever, L’Oreal, Eli Lilly, Dell, and Netflix [1, 3]. Regarding market revenues, a recent report shows that the enterprise crowdsourcing market grew 53 % in 2010, 75 % in 2011, and was expected to double in 2012 [4]. Likewise, crowdsourcing has been expanding to different fields including software development [5], marketing [6] and hospitality [7].

As a response to this popularity, organisations are struggling to assimilate and standardise business processes around this strategy, a movement that has been coined Business Process Crowdsourcing (BPC) [8, 9]. BPC can be seen as *a traditional set of organisational activities done by crowdsourcing entities, plus the coordination of the entire business process*. By establishing BPC, organisations can integrate the crowdsourcing strategy with their day-to-day business processes, being “able to seamlessly bring together the crowd, individual actors, and the machine” [10]. Thus, it enables incorporating the crowdsourcing capabilities within the organisational value proposition [11].

Although the advantages of crowdsourcing to organisations have already been highlighted by several researchers [8, 12], only recently have there been noticeable efforts researching BPC [10, 11]. Even though they investigate the BPC phenomenon from different angles, these studies consistently suggest that, in the long run, BPC needs to be established as a continuous organisational process, which requires systematic management of the strategy. Aligning with these efforts, we have conducted a 3-year research project that focused on BPC from a managerial decision-making perspective. We started the project by reviewing the existing crowdsourcing literature and eliciting the main BPC concepts, activities and contextual factors. We then articulated all these elements into a decision framework consisting of three phases: decision to crowdsource, process design, and system configuration [9].

Based on this framework, the project then investigated how to support managerial decisions in each phase. More precisely, we articulated the several factors, relationships, decision choices, and recommendations suggested by existing literature in the decision framework. In this way, the project analysed and conceptualised the decision to crowdsource [13] and the various design and configuration options [9]. Besides conceptualisation, we have also developed a more formal BPC ontology, which consists of more than 100 domain concepts, relationships and rules [11]. The ontology itself highlights the complexity inherent in establishing a BPC process.

The next logical step in our research consists of helping decision makers—project managers, business analysts, and process designers—making analytical decisions in the crowdsourcing establishment. This type of support is within the typical domain of Decision Support Systems (DSSs) [14, 15]. As a part of our research project, the current study aims specifically at developing a decision tool supporting the establishment of BPC. Given the above discussion, the tool should be beneficial by supporting managers on not only the decision to crowdsource or not [13], but also the various inconspicuous decisions that follow the decision to crowdsource, which include design and configuration issues [10]. Building on the BPC ontology, the tool emphasises strategic decisions, extending the managers’ capability to make informed decisions about the entire BPC process.

Considering the impact of Design Science on DSSs [14], our study follows a Design Science paradigm [16, 17]. In particular, this paper reports the development of a crowdsourcing decision tool, viewed as a design artefact. To provide a solid knowledge base for building this artefact, the study relies upon the BPC ontology previously developed by the project [11]. This knowledge base is integrated with the tool’s architecture. By doing so, the tool consolidates existing research knowledge in a structured decision-making process.

The current study should benefit both practitioners and academics. From a practical point of view, the study provides a computer-based tool supporting organisations in establishing crowdsourcing strategies. From an academic point of view, the tool investigates the establishment of BPC at the concrete decision level, and thus complements prior conceptual efforts [11, 18]. Furthermore, since the tool is based on a BPC ontology [11], its development responds to the call for a more integrated and holistic view on crowdsourcing research [19, 20].

2 Literature Review

2.1 Identification of Problems

The concept of crowdsourcing was first introduced by Howe [2] in 2006. By that time, researchers discussed and explored what the concept means and its potential applications [6, 19]. These efforts contributed to an initial conceptualisation of crowdsourcing, usually referred to as a process utilising the members of the crowd and the Internet with the purpose to fulfil ad hoc tasks. They also discussed the application of crowdsourcing strategies in several areas including information processing, idea gathering, design [21, 22], and supporting decision making [23].

From an organisational point of view, crowdsourcing may consist of regular activities performed by internal employees and ad hoc activities performed by the external crowd [1]. Thus, there is a need to seamlessly integrate these activities into an organisational workflow or BPC [8, 9]. Such integration helps organisations become more efficient, as pointed out by Tranquillini et al. [10]. By integrating crowdsourcing processes in existing business processes, they can be built on top of existing business process management (BPM) technology and information systems [10]. Furthermore, this integration helps crowdsourcing to become a more mature technology for organisations to exploit [24, 25].

However, the establishment of BPC is not a straightforward task. The existing literature highlights several issues and challenges related to this establishment [5, 22]. For instance, Djelassi and Decoopman [22] suggested that it is not a simple, but a rather complex process. These authors viewed the process as requiring the coordination of several business components, including infrastructure, incentive mechanisms, the crowd, customers, and also the financial viability. In a similar vein, Tranquillini et al. [10] identified a variety of options and configurations for BPC integration. Recently, Thuan et al. [11] synthesised the components, processes, activities, and data entities necessary for this integration from an ontological point of view. They noted the diversity of related concepts, hierarchical relationships, decision-making relationships, and business rules related to BPC. Given this complexity, a critical challenge is how to help organisations establishing BPC.

DSSs help organisations making decisions about wicked problems like BPC. In the crowdsourcing field, a few exploratory DSSs have been developed. Geiger and Schader [20] proposed a foundation for constructing a recommendation system matching individuals in the crowd with types of crowdsourcing tasks. Recently, Prokesch and Wohlenberg [26] developed a DSS that processes results from the crowd. Although

these systems can support certain aspects of crowdsourcing, they are mainly focused on very specific functions like task assignment [20] and results aggregation [26], rather than the whole integrated process. Consequently, there is still a need for a DSS tool supporting the entire BPC process. From the discussion in this section and the introductory section, we note that such tools should accomplish the following requirements:

- Assist managers deciding how to establish a BPC strategy or not. This assistance should be given as guidelines and recommendations.
- Build a comprehensive, integrated view of BPC. In other words, the tool should support the integrated BPC process, not individual activities. Several DSS studies suggest that such an integrated view can be achieved by using sound domain ontologies [27].
- Support micro-decisions related to the BPC process, including process design and configuration. Within each component, the (sub) issues, their alternatives and guidance to choose among these alternatives should be specified.
- Provide a means for the effective processing and presenting of knowledge related to the establishment of BPC.

2.2 DSS View

Decision Support Systems is a research area with a long history in Information Systems (IS), which can be traced back to Simon's intelligence-design-choice model developed in 1960 [15]. In this research area, the focus is on supporting and improving decision-making for wicked, normally semi-structured and unstructured decisions [14]. The term 'support' is important in DSSs, since these systems are not meant to replace decision makers, but help them extend their capabilities and make more informed, better decisions [15]. Normally, this support requires integrating domain models conceptualising the application domains, which helps decision makers to understand and explore different decision options.

Due to the long history, a large number of DSSs have been studied and developed in IS and its related fields for various endeavours [14, 28]. To structure these systems, several taxonomies have been proposed. Power [28] suggested five types of DSSs including data driven, model driven, knowledge driven, document driven, and communication driven, whose names reflect the main foundation backing the DSS. Recently, Arnott and Pervan [14, 29] analysed the DSS literature and developed a seven-type taxonomy, which was based on four dimensions: dominant technology, theory foundations, targeted users, and decision tasks. Using these dimensions, they suggested classifying DSSs into: (1) personal DSSs for individual managers; (2) group DSSs for a group of decision makers; (3) negotiation support systems, which are group support systems but involve negotiation functions; (4) intelligent DSSs, using artificial intelligence; (5) knowledge DSSs, which provide knowledge storage, retrieval, transfer, and application; (6) data warehousing, processing large-scale (big) data for decision support; and (7) enterprise reporting and analysis systems. Positioning our tool in these landscapes, we note our work is a personal model-driven DSS, since we focus on supporting independent managers and base the decision support on a BPC ontology.

Despite the variety of DSSs, the generic DSS architecture seems quite consistent. By and large, Holsapple [30] suggested four main DSS components: language component, presentation component, knowledge component, and problem-processing component. The language component processes user inputs. The problem-processing component tries to identify, analyse, and model the problem, which provides information, alternatives and advice for addressing the problem. This process is based on the knowledge component, which stores knowledge related to the problem. The output from the DSSs is presented to decision makers by the presentation component. Şeref and Ahuja [31] proposed a similar architecture grouping the language and presentation components into a graphical user interface (GUI), and divided the problem-processing component into model and database aspects. Since the components proposed by Holsapple [30] seem to clearly separate the major concerns, we adapted this schema to structure the proposed tool, which is discussed in Sect. 4.

3 Research Overview

The tool development follows the Design Science paradigm [16, 17], which has been adopted by many DSS developments. The links between DSSs and Design Science have been highlighted by prominent academics in both fields. In DSSs, Arnott and Pervan [32] argue that most DSS developments somehow correspond to what Hevner et al. [17] define as a design artefact. Even Hevner et al. [17] in their seminar paper demonstrated Design Science using three artefacts, of which two are DSSs. In Design Science, research mainly aims at developing artefacts solving wicked problems [16, 33]. Hevner and Chatterjee [16] demonstrate that development processes should be simultaneously relevant and rigorous. For the current study, while the tool's relevance has already been clarified and discussed in the literature review section, it is more challenging to fully demonstrate rigor.

Since crowdsourcing is an emerging field [1, 20], there are (at least) three challenges related to rigor. First, having recently conducted a review of crowdsourcing literature [9], we could not find a prevailing crowdsourcing theory that could be used as a rigorous knowledge base for development. Second, like other emerging and very dynamic research fields, different and sometimes conflicted findings can be found in the crowdsourcing research literature [20], which affects any attempts to fully justify every step of the tool development. These two challenges lead to the third one, which is related to the artifact's internal validity. More precisely, it is challenging to describe concepts formally and demonstrate logical assertions when a common understanding of BPC is still lacking, a dominant theory is missing, and the field is immature.

The current study addresses these challenges from two perspectives: a knowledge base perspective and a software development perspective. Regarding the former, we recognise that further consolidation of domain knowledge is necessary. Multiple researchers have suggested that domain ontologies help with constructing and consolidating domain knowledge [34, 35]. More precisely, they posit that, as conceptual modelling techniques, ontologies may formalise the domain and ease communication among different parties [36, 37]. While agreeing with these, we suggest also considering ontologies in the context of DSSs. In DSSs, Miah and von Hellens [27] used

ontologies as knowledge components for “structuring and representing problem specific knowledge into a knowledge repository”. Aligning with these authors, we developed our tool based on a BPC ontology [11].

From the software development perspective, an appropriate software development method is necessary to deal with the immature nature of the field. The current study follows Lim et al.’s [38] suggestion to adopt a rapid prototyping method. This method deals with complexity through iterative development and revision of a few prototypes [39], and allows traversing the tool’s design space [38]. Prototyping is appropriate for DSS development, as suggested by Miah et al. [40] regarding the development of an expert system supporting rural business operators, and Antunes et al. [41] regarding the development of a decision tool supporting geo-collaboration.

In short, the ontological approach and prototyping method both help with consolidating the domain knowledge and iteratively understanding and developing the tool. These two perspectives are further detailed in the following sections, where we describe the tool’s architecture and development details.

4 Tool Architecture

The tool’s architecture is based on the components proposed by Holsapple [30] (Literature Review section). As shown in Fig. 1, the architecture has three main components: GUI, problem processing component, and knowledge component. The GUI component enhances the interaction between the tool and its users, i.e. managers and process designers, who make decisions on adopting and designing BPC processes. This component accepts parameters from the users. It also helps validating them by providing explanations about the parameters drawn from the domain knowledge and ontology. These inputs are processed by the problem process component, where parameters are used to formulate the problem and the associated context. Furthermore, the problem process component controls input flows by choosing and adapting what elements the GUI presents. It also manipulates data entries based on the knowledge component.

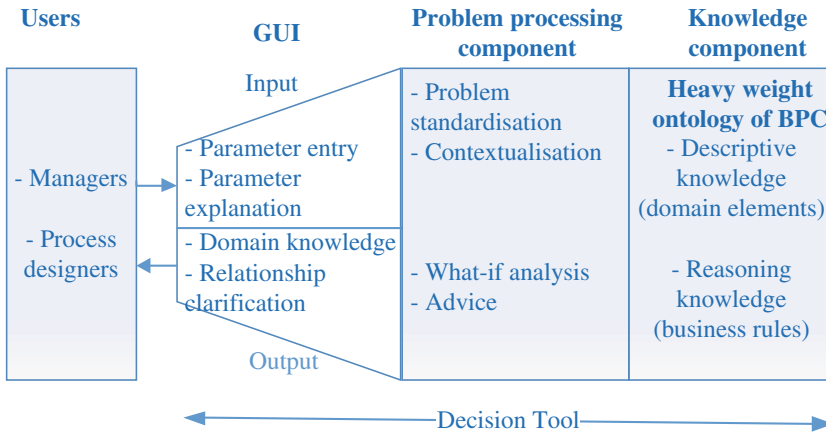


Fig. 1. Tool’s architecture (adapted from [30])

The knowledge component adopts an ontology built in our previous research. Figure 2 presents a lightweight view of the BPC ontology. The heavyweight ontology and its details can be found in [11]. As presented in Fig. 2, the knowledge component consists of descriptive and reasoning knowledge. Regarding descriptive knowledge, it provides definitions and descriptions of concepts that have to be considered in the decision-making process. It also includes a hierarchy of relationships among the (sub) concepts (presented as ‘include’ and ‘categorise’ relationships in Fig. 2). Reasoning knowledge provides business rules constraining these ontological elements.

Using the knowledge component, decision-makers can perform what-if analysis by comparing the knowledge specified by the ontology with the expressed input parameters. In this way, the ontology serves as a knowledge base capturing the basic profiles of crowdsourcing projects, which can be adapted based on project conditions and intervention plans. Through this adaption, the decision tool can detect ontological

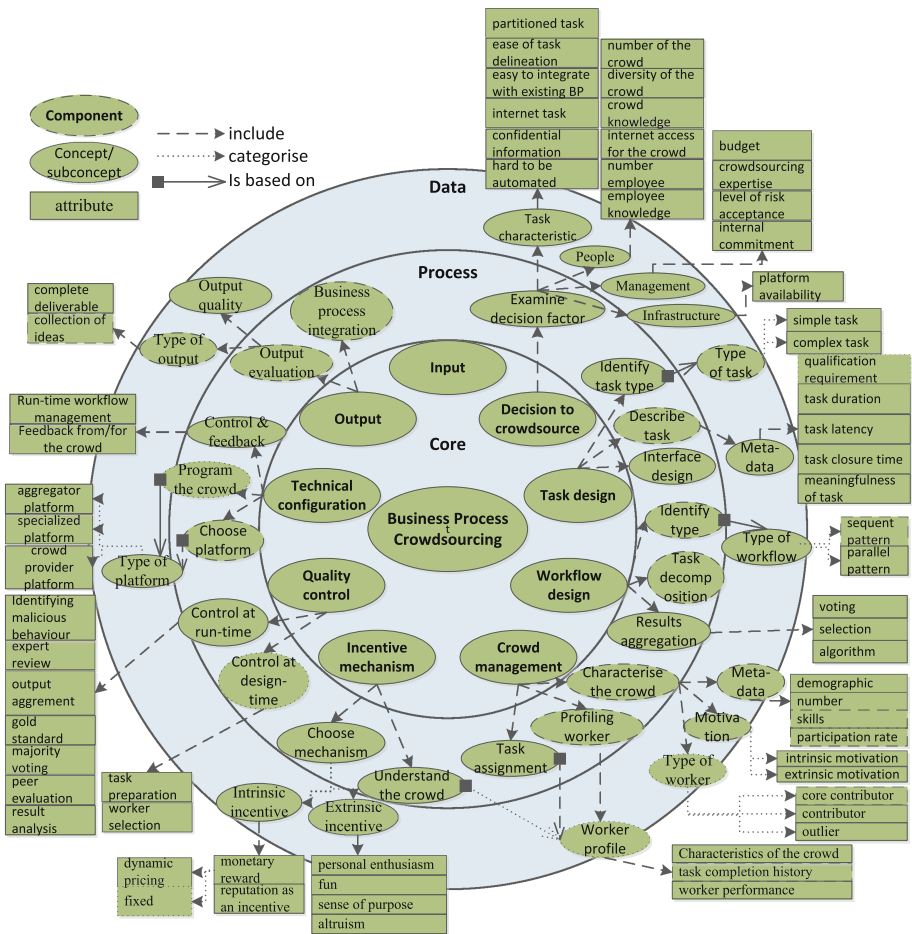


Fig. 2. A lightweight ontology of BPC [11]

inconsistencies in the available data of the crowdsourcing project. As a result, it provides advice on how to set up a crowdsourcing strategy for a particular organisational context, and configure the process details, which in turn are presented as GUI's outputs.

5 Tool Development

Following the rapid prototyping method, tool development consisted of two phases: spreadsheet-based DSS and web-based DSS. Şeref and Ahuja [31] suggest that spreadsheets are useful tools for modelling and developing DSS. In the current work, the spreadsheet development demonstrates domain knowledge articulation, transferring knowledge drawn from the ontology into computer-based formulations. Figure 3 shows the spreadsheet component supporting the decision to crowdsource or not. In spite of visual austerity, the prototype implements all three architectural components. A question and answer section gathers parameters about the BPC organisational context. These parameters are then processed and transferred to a back-end sheet where the ontology elements are applied. This back-end sheet implements the knowledge component. After knowledge processing, several recommendations are provided by the tool (the 'Advice' area in Fig. 3). Besides Fig. 3, the spreadsheet tool also has another sheet that supports BPC process design and configuration.

A	B	C	D	E	F	G
	#	Decision category	Decision factor	Questions		Answer
2	1	Task	Internet vs. Physical	The task and its input/output can be delivered and collected through the internet		<input type="radio"/>
3	2		Integration with existing BP	Crowdsourcing could be integrated with the existing organisational business processes		<input type="radio"/>
4	3		Interactive	The task requires frequent interaction and communication between the organisation and the crowd, or between the members of the crowd		<input checked="" type="checkbox"/>
5	4		Delineation	The crowdsourcing task should be well-defined		<input checked="" type="checkbox"/>
6	5		Confidential information	The task includes confidential information, including privacy and intellectual property consideration		<input type="radio"/>
7	6		Partitionable	The task can be partitionable into smaller pieces of work		<input type="radio"/>
8	7	People	The crowd for task	There are high numbers of crowd members for perform crowdsourcing tasks		<input checked="" type="checkbox"/>
9	8		Employee for task	The organisation/project has too few internal employees to deploy the task		
10	9	Management	Budget	Budget allocated for the crowdsourcing project is sufficient		
11	10		Crowdsourcing experts	The organisation/project has appropriate expertise and experience to coordinate the crowdsourcing activities		
12	11		Level of risk acceptance	The organisation has high level of acceptance related to risks, e.g. low quality results and loss of intellectual property		
13	12	Environment	Internal commitment	Internal employees have low commitment to crowdsourcing		
14	13		Platform	There are high availability of crowdsourcing platforms that can deploy the crowdsourcing activities		
15						
16						
17		Advice	Crowdsourcing task with additional actions: - Clearly define task in the latter stages of the crowdsourcing process - Define tasks hiding confidential information.			

Fig. 3. Spreadsheet-based tool on the decision to crowdsource

Using this spreadsheet, we performed several what-if analyses generating a range of probable outcomes of the BPC project. This type of analysis allowed us to review and adjust the ontology implementation. Of course this prototype had its own disadvantages,

especially regarding the limited utilisation of the knowledge base, and in particular the difficulties navigating between the decision to crowdsource and BPC design and configuration. The web-based prototype addresses these concerns.

5.1 Web-Based Prototyping

The web-based prototype was developed as an improved and revised version of the spreadsheet prototype. This prototype, which was implemented using Php and MySQL, provides wider access to the knowledge base. The entity-relationship model (implemented with MySQL) enables more systematic information management. Furthermore, data structures were added to support project management. For instance, a user can create multiple BPC projects, each of which supports a particular organisational context and a particular phase of BPC. The database structure is presented in Fig. 4.

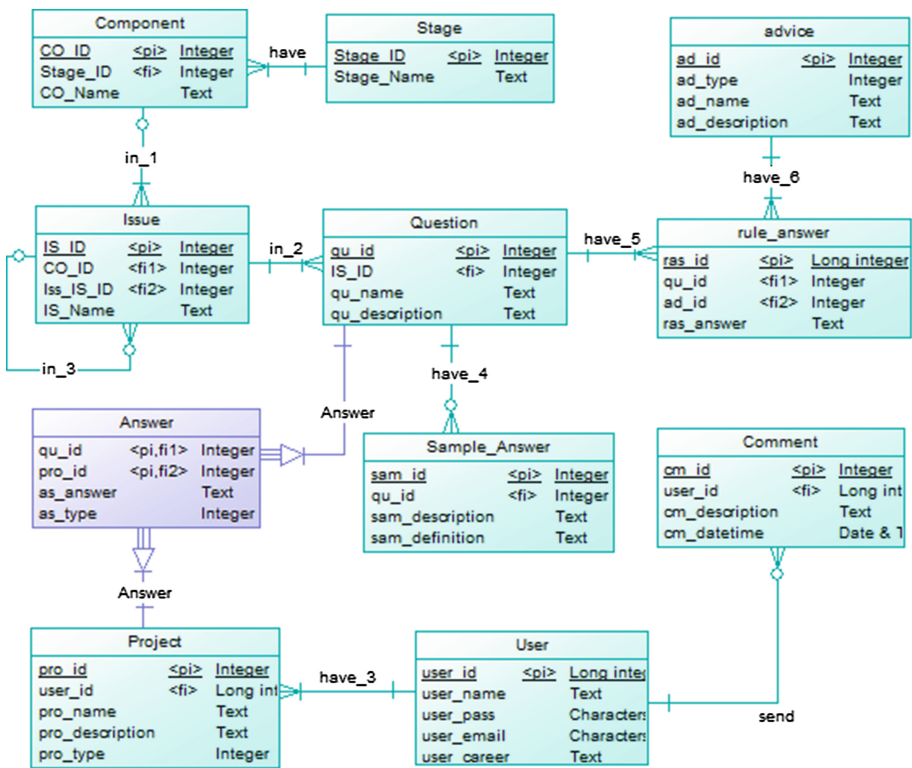


Fig. 4. Data structure of the web-based prototype

The prototype consists of two decision functions, supporting the decision to crowdsource (Fig. 5) and design process of BPC establishment (Fig. 6). More precisely, the former provides a check list of decision factors and analytical advice on

making the decision to crowdsource. The latter specifies the design process of BPC, which helps to organise its establishment in an appropriate structure. To keep the prototype consistent, the user-interfaces of the two functions are consistently designed and organised in five areas (Figs. 5 and 6). The right-hand-side is dedicated to provide an overview of the decision to crowdsource and BPC design process. The left-hand side is divided into four areas with inputs and outputs. In the input area, the tool allows users to choose a design issue. After choosing a particular issue, an explanation and a pre-defined parameter are presented. If the user changes the parameter, advice will be provided. This prototype is currently being tested.

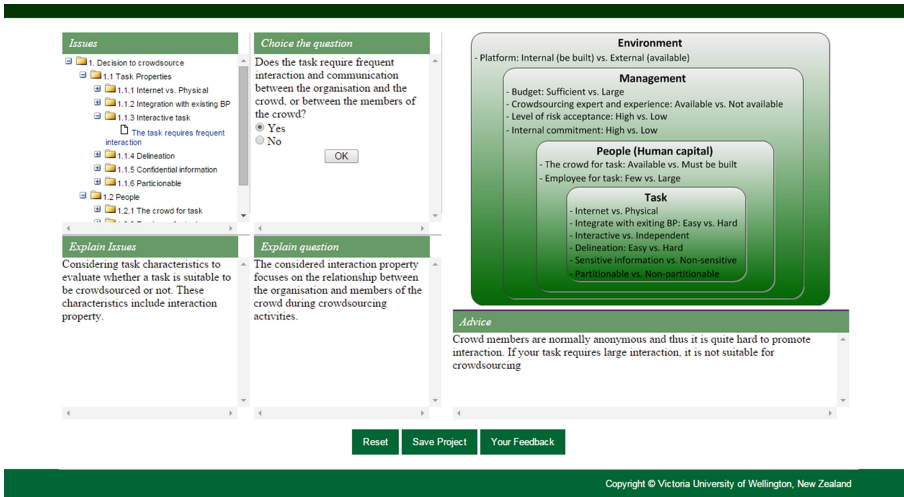


Fig. 5. A web-based tool: a screenshot on the decision to crowdsource

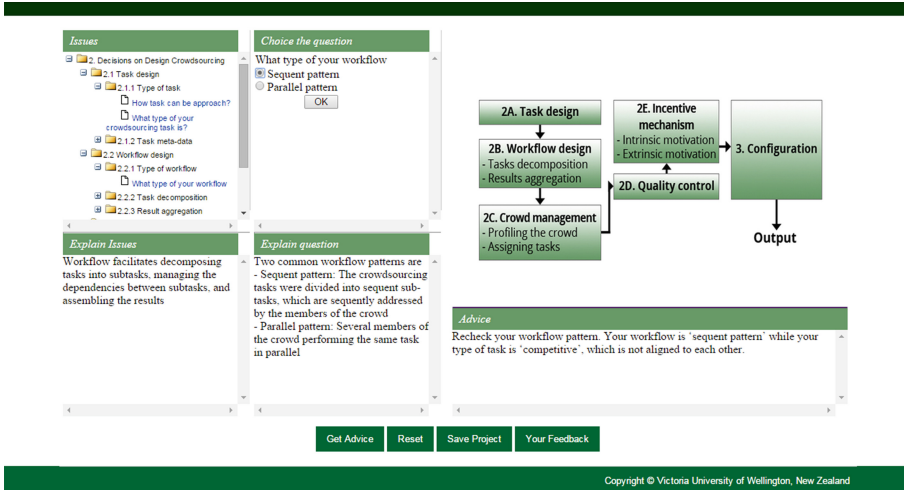


Fig. 6. A web-based tool: a screenshot on the process design of BPC establishment

6 Discussion and Conclusion

DSSs may help organisations adopting and configuring novel, complex business processes like crowdsourcing. Adopting the Design Science paradigm, this study developed a DSS supporting managers addressing the complexity of BPC projects [8, 12, 42]. Unlike the existing decision tools in the crowdsourcing domain [20, 26], which focus on individual aspects of BPC, the current study adopted an ontological approach for developing the DSS. As a result, the developed tool can support the whole, integrated BPC project, from the decision to crowdsource or not to process design and configuration.

Considering the emerging nature of the field [1], which increases the complexity of developing such a tool (especially demonstrating rigor), a prototyping development method was chosen [39]. We developed the tool architecture consisting of three main components: GUI, information processing component, and knowledge component. This architecture was utilised in two prototype implementations: spreadsheet-based and web-based. While the spreadsheet-based prototype allowed us to experiment with several crowdsourcing scenarios and analysing the parameters and recommendations provided by the ontology, the web-prototype is targeted to project and business managers, and process designers, who make managerial decisions in organisational contexts. Thus, the two prototypes make complementary contributions to research and practice.

Our research contributes to the current knowledge by building a decision tool for BPC, and thus validating an ontology of BPC [11]. More precisely, it structures concepts, relationships, business rules, and what-if scenarios of BPC into two computer-based prototypes. From a Design Science perspective, implementing these prototypes is an evaluation form of the ontology [43], which demonstrates the applicability of the ontology. From a crowdsourcing research perspective, the prototypes can be seen as a further contribution to the conceptualisation and standardisation of BPC. Since the prototypes allow researchers to explore different BPC scenarios, they can also be used as a research tool.

In future work, we will involve managers in using the prototypes with the purpose to further validate the usefulness of the ontology and tool. We can also evaluate the tool by conducting comparative experiments. In the experiments, participants may be asked to make crowdsourcing decisions in specific scenarios. One group of participants will make decisions using the tool and the other will make decision without the tool. Comparing performance of the two groups can provide empirical evaluation on the usefulness of the tool. From a system development perspective, our prototypes may be integrated with the work by Tranquillini et al. [10], which considered more technical details about BPC implementation. Thus, another interesting research direction is to investigate how to connect the managerial and technical domains. This connection would provide a system supporting organisations from the time they decide to crowdsource until the time they instantiate a BPC process on a particular crowdsourcing platform.

References

1. Zhao, Y., Zhu, Q.: Evaluation on crowdsourcing research: current status and future direction. *Inf. Syst. Front.* **16**(3), 417–434 (2014)
2. Howe, J.: The rise of crowdsourcing. In: *Wired Magazine* 2006, pp. 1–4. Dorsey Press, Homewood (2006)
3. Rosen, P.A.: Crowdsourcing lessons for organizations. *J. Decis. Syst.* **20**(3), 309–324 (2011)
4. Massolution: The Crowd in the Cloud: Exploring the Future of Outsourcing. Massolution (2013)
5. Zogaj, S., Bretschneider, U., Leimeister, J.M.: Managing crowdsourced software testing: a case study based insight on the challenges of a crowdsourcing intermediary. *J. Bus. Econ.* **84**(3), 375–405 (2014)
6. Whitla, P.: Crowdsourcing and its application in marketing activities. *Contemp. Manage. Res.* **5**(1), 15–28 (2009)
7. Brabham, D.C., et al.: Crowdsourcing applications for public health. *Am. J. Prev. Med.* **46**(2), 179–187 (2014)
8. La Vecchia, G., Cisternino, A.: Collaborative workforce, business process crowdsourcing as an alternative of BPO. In: Daniel, F., Facca, F.M. (eds.) *ICWE 2010. LNCS*, vol. 6385, pp. 425–430. Springer, Heidelberg (2010)
9. Thuan, N.H., Antunes, P., Johnstone, D.: Toward a nexus model supporting the establishment of business process crowdsourcing. In: Dang, T.K., Wagner, R., Neuhold, E., Takizawa, M., Küng, J., Thoai, N. (eds.) *FDSE 2014. LNCS*, vol. 8860, pp. 136–150. Springer, Heidelberg (2014)
10. Tranquillini, S., et al.: Modeling, enacting, and integrating custom crowdsourcing processes. *ACM Trans. Web (TWEB)* **9**(2), 7 (2015)
11. Thuan, N.H., et al.: Building an enterprise ontology of business process crowdsourcing: a design science approach. In: *PACIS 2015 Proceedings*, Paper 112 (2015)
12. Khazankin, R., Satzger, B., Dustdar, S.: Optimized execution of business processes on crowdsourcing platforms. In: *IEEE 8th International Conference on Collaborative Computing: Networking, Applications and Worksharing*, Pittsburgh, PA (2012)
13. Thuan, N.H., Antunes, P., Johnstone, D.: Factors influencing the decision to crowdsource: a systematic literature review. *Inf. Syst. Front.* 1–22 (2015)
14. Arnott, D., Pervan, G.: A critical analysis of decision support systems research revisited: the rise of design science. *J. Inf. Technol.* **29**(4), 269–293 (2014)
15. Hosack, B., et al.: A look toward the future: decision support systems research is alive and well. *J. Assoc. Inf. Syst.* **13**(5), 315–340 (2012)
16. Hevner, A., Chatterjee, S.: In: Sharda, R., Voß, S. (eds.) *Design Research in Information Systems: Theory and Practice. Integrated Series in Information Systems*, vol. 22. Springer, Heidelberg (2010)
17. Hevner, A., et al.: Design science in information systems research. *MIS Q.* **28**(1), 75–105 (2004)
18. Hetmank, L., Developing an ontology for enterprise crowdsourcing. In: *Multikonferenz Wirtschaftsinformatik, Paderborn*, pp. 1089–1100 (2014)
19. Estellés-Arolas, E., González-Ladrón-de-Guevara, F.: Towards an integrated crowdsourcing definition. *J. Inf. Sci.* **38**(2), 189–200 (2012)
20. Geiger, D., Schader, M.: Personalized task recommendation in crowdsourcing information systems—current state of the art. *Decis. Support Syst.* **65**, 3–16 (2014)
21. Leimeister, J.M., et al.: Leveraging crowdsourcing: activation-supporting components for IT-based ideas competition. *J. Manage. Inf. Syst.* **26**(1), 197–224 (2009)

22. Djelassi, S., Decoopman, I.: Customers' participation in product development through crowdsourcing: issues and implications. *Ind. Mark. Manage.* **42**(5), 683–692 (2013)
23. Chiu, C.-M., Liang, T.-P., Turban, E.: What can crowdsourcing do for decision support? *Decis. Support Syst.* **64**, 40–49 (2014)
24. McCormack, K., et al.: A global investigation of key turning points in business process maturity. *Bus. Process Manage. J.* **15**(5), 792–815 (2009)
25. Van Looy, A., et al.: Choosing the right business process maturity model. *Inf. Manage.* **50**(7), 466–488 (2013)
26. Prokesch, T., Wohlenberg, H.: Results from a group wisdom supporting system. In: *Proceedings of the European Conference on Information Systems (ECIS) 2014*, Paper 7 (2014)
27. Miah, S., Kerr, D., von Hellens, L.: A collective artefact design of decision support systems: design science research perspective. *Inf. Technol. People* **27**(3), 259–279 (2014)
28. Power, D.J.: Decision support systems: a historical overview. In: *Handbook on Decision Support Systems 1*, pp. 121–140. Springer, Heidelberg (2008)
29. Arnott, D., Pervan, G.: Eight key issues for the decision support systems discipline. *Decis. Support Syst.* **44**(3), 657–672 (2008)
30. Holsapple, C.W.: DSS architecture and types. In: *Handbook on Decision Support Systems 1*, pp. p. 163–189. Springer, Heidelberg (2008)
31. Şeref, M.M., Ahuja, R.K.: Spreadsheet-based decision support systems. In: *Handbook on Decision Support Systems 1*, pp. 277–298. Springer, Heidelberg (2008)
32. Arnott, D., Pervan, G.: Design science in decision support systems research: an assessment using the Hevner, March, Park, and Ram guidelines. *J. Assoc. Inf. Syst.* **13**(11), 923–949 (2012)
33. Pries-Heje, J., Baskerville, R.: The design theory nexus. *MIS Q.* **32**(4), 731–755 (2008)
34. Osterwalder, A.: The business model ontology: a proposition in a design science approach. *Institut d'Informatique et Organisation*. Lausanne, Switzerland, University of Lausanne, Ecole des Hautes Etudes Commerciales HEC (2004)
35. Ostrowski, L., Helfert, M., Gama, N.: Ontology engineering step in design science research methodology: a technique to gather and reuse knowledge. *Behav. Inf. Technol.* **33**(5), 443–451 (2014)
36. Guarino, N., Oberle, D., Staab, S.: What is an ontology? In: Staab, S., Studer, R. (eds.) *Handbook on ontologies*, pp. 1–17. Springer, Heidelberg (2009)
37. Valaski, J., Malucelli, A., Reinehr, S.: Ontologies application in organizational learning: a literature review. *Expert Syst. Appl.* **39**(8), 7555–7561 (2012)
38. Lim, Y.-K., Stolterman, E., Tenenberg, J.: The anatomy of prototypes: prototypes as filters, prototypes as manifestations of design ideas. *ACM Trans. Comput. Hum. Interact. (TOCHI)* **15**(2) (2008)
39. Kordon, F.: An introduction to rapid system prototyping. *IEEE Trans. Softw. Eng.* **28**(9), 817–821 (2002)
40. Miah, S.J., Kerr, D.V., Gammack, J.G.: A methodology to allow rural extension professionals to build target-specific expert systems for Australian rural business operators. *Expert Syst. Appl.* **36**(1), 735–744 (2009)
41. Antunes, P., et al.: Integrating decision-making support in geocollaboration tools. *Group Decis. Negot.* **23**(2), 211–233 (2014)
42. Vukovic, M.: Crowdsourcing for enterprises. In: *2009 World Conference on Services-I*. IEEE, Los Angeles, CA (2009)
43. Peffers, K., Rothenberger, M., Tuunanen, T., Vaezi, R.: Design science research evaluation. In: Peffers, K., Rothenberger, M., Kuechler, B. (eds.) *DESRIST 2012*. LNCS, vol. 7286, pp. 398–410. Springer, Heidelberg (2012)

Finding Similar Artists from the Web of Data: A PageRank Based Semantic Similarity Metric

Phuong T. Nguyen¹(✉) and Hong Anh Le²

¹ Institute of Research and Development, Duy Tan University, Da Nang, Vietnam
phuong.nguyen@duytan.edu.vn

² Hanoi University of Mining and Geology, Hanoi, Vietnam
lehonganh@hmg.edu.vn

Abstract. Since its commencement, the **Linked Open Data** cloud has been quickly become popular and offers rich data sources for quite a number of applications. The potential for application development using **Linked Data** is immense and needs intensive research efforts. Until now, the issue of how to efficiently exploit the data provided by the new platform remains an open research question. In this paper we present our investigation of utilizing a well-known encyclopedic dataset, **DBpedia** for finding similar musical artists. Our approach exploits a PageRank based semantic similarity metric for computing similarity in **RDF** graph. From the data provided by **DBpedia**, the similarity results help find out similar artists for a given artist. By doing this, we are also be able to examine the suitability of **DBpedia** for this type of recommendation tasks. Experimental results show that the outcomes are encouraging.

Keywords: Linked Open Data · Personalized PageRank · Semantic similarity

1 Introduction

The deployment of the **Linked Open Data** brings in a rich data source containing a broad range of knowledge, spanning from life science, environment, industry to entertainment. This forms the so called **Web of Data** which offers a greater convenience and fosters the development of **Semantic Web** applications. To date, information sharing, information retrieval [15, 16], community detection, recommendation systems [13, 14] - to name a few - are noteworthy applications that successfully leverage **Linked Data**. Recommender systems are built to suggest a category of things, eg. books, movies, songs to a user using different sources of background data [13]. By exploiting **Linked Data**, recommender systems are able to enrich their background data, thereby providing users with meaningful recommendations. Due to their features, one of the main tasks of recommender systems is to find similar entities with a given resource or to evaluate to which extent two resources are alike. This leads to the problem of computing similarity between semantically related resources. For music recommender systems, this

is highly beneficial since they can exploit this function to find similar artists for a given singer or music band using **Linked Data**. The list suggested by the systems helps listeners browse for more artists and discover other masterpieces that may belong to their favorites.

The proliferation of **Linked Data** source facilitates the development of recommender systems, albeit posing some thorny issues. Since data is freely processed and uploaded by various parties, it might be neither complete, nor consistent. Furthermore, a dataset may be suitable for some tasks but not all tasks. This raises concerns over *fitness for use* [12, 19], since data quality has an immense influence over application performance. As a result, assessing quality for **Linked Data** is of wide interest across the research community. In this paper, we investigate the ability of a semantic similarity metric based on PageRank for finding similar artists from an encyclopedic data source **DBpedia**¹. By doing this, at the same time we are able to evaluate the *fitness for use* of **DBpedia** for this type of recommendation tasks. Our experimental evaluations are conducted in accordance with a Last.fm dataset for musical artists recommendation. Experimental results demonstrate interesting outcomes.

Our contributions in this paper are summarized as follows:

- Investigating the possibility of using Personalized PageRank for measuring semantic similarity in **Linked Data**.
- Evaluating the *fitness for use* of **Linked Data** in the music domain for recommendation tasks.

The paper is presented in the following structure. Section 2 brings an overview of related work. Section 3 reviews PageRank, the ranking algorithm used by the Google search engine. Section 4 introduces a featured-based semantic similarity using Personalized PageRank. The experimental results are clarified in Sect. 5. Finally, some conclusions and future work are discussed in Sect. 6.

2 Related Work

A system leveraging **Linked Data** for recommending music is introduced in [13]. An algorithm for computing semantic similarity in RDF graph, the **Linked Data Semantic Distance** algorithm or **LDS** is proposed to calculate similarity between a piece of music or an artist and a set of candidates. Based on the filtering results, the recommender system produces a list of songs or artists that is then presented to users. A hybrid recommender system based on unified Boltzmann machines has been presented in [18] to deal with the problems of cold-start and lower accuracy in recommender systems. The system models item interactions and learns weights representing the importance of different pairwise interactions by integrating collaborative and content information. Based on the probabilistic models, the system can predict whether a user will act on a specific item.

¹ <http://dbpedia.org>.

Recently, the problem of evaluating **Linked Data** quality has attracted significant attention from the research community. Zaveri *et al.* introduce a comprehensive survey on data quality assessment in [10]. The paper presents a systematic review of the most notable approaches dealing with **Linked Data** quality. This work can be considered as a clue to facilitate the development of methodologies for analyzing **Linked Data** quality. So far, quite a number of schemes to analyze data quality have been developed. In [11], a quality assessment methodology consisting of three phases and six steps is introduced. Alongside procedures for identifying quality dimensions, the approach provides some proposals for refining data. With the involvement of human beings for evaluating data quality, automated, semi-automated processes are utilized to identify and finally fix the problem concerning data quality. Based on the observation that most **Linked Data** quality assessment methodologies either depend largely on manual configurations or lack scalability, Kontokostas *et al.* devise a methodology to assess the quality of **Linked Data** resources in [19]. The methodology is derived from test-driven software development. It incorporates ontologies, vocabularies and knowledge bases into quality assessment. This aims at obtaining automation and scalability at once. The approach is expected to boast more advantages compared to the existing ones. There exist some software tools for **Linked Data** quality assessment, such as *Sieve* which is presented in [17]. This software has been integrated into an existing framework for data integration, *LDIF* [20]. Based on pre-defined scoring functions, *Sieve* can generalize a set of quality indicators from input metadata. It can then fuse data from different **Linked Data** sources to meet user requirements.

3 PageRank

In order to rank Web pages properly, a crucial task for search engines is to measure the level of importance of every Web page involved in the indexing process. One of the key components making up the Google search engine is the mechanism for ranking Web pages, the PageRank algorithm [3,5]. The mechanism models a random Web surfer who visits Web pages by clicking on the links. By PageRank, the Web is represented as a directed graph with n nodes and each node has a rank associated with it. The rank of a node i corresponds to the probability that the Web surfer ends up visiting the node. A rank is calculated according to the relationship with its neighborhood. A node gets an amount of rank from every node pointing to it and in turn conveys its rank to the nodes it refers to. One node is about to have a high rank if it is referenced by nodes with high rank.

To compute the rank of nodes, an $n \times n$ transition matrix G is built whereby row i^{th} represents the rank that node i^{th} transfers to other nodes that it has links to. If node i has l_i outbound links then it transfers $rank_i = \frac{1}{l_i}$ to all of its neighborhood nodes. In the transition matrix, the cell at row i^{th} and column j^{th} has the value of $rank_i$ if there is a link from node i to node j , otherwise it has the value of 0. From this definition, however, there is a problem with *dangling nodes*,

i.e. those with no outgoing links. By these nodes, the PageRank vectors degrade very quickly and produce inappropriate ranks. By the Web surfer phenomenon, if the surfer visits dangling nodes, he will be stuck there and never be able to move out. To circumvent this, two vectors are introduced, thereby re-phrasing the transition matrix as follows:

$$G' = G + \delta.\omega$$

in which $\omega = (\omega_1 \ \omega_2 \ .. \ \omega_n)$ and $\sum_{i=1}^n \omega_i = 1$, in general all entries ω_i are set to $\frac{1}{n}$; δ is a column vector with entry i^{th} $\delta_i = 1$ if i is a dangling node and $\delta_i = 0$ otherwise. The introduction of ω and δ replaces all entries 0 of a dangling node in the transition matrix with $\frac{1}{n}$. This imposes the dangling nodes virtual link to all the other nodes with the same probability. Based on the original transition matrix, the following one is defined to cover all probable activities of a Web surfer [4]:

$$G'' = d.G' + (1 - d)v$$

in which d is the damping factor ($0 \leq d < 1$); v is the personalization vector. The parameter $(1 - d)$ represents the probability that the surfer jumps to Web pages by other means rather than by clicking on the links, e.g. by keying Web pages' URLs in the address bar of his browser. The PageRank vector contains the ranks of all nodes, i.e. entry i^{th} holds the rank of the i^{th} node in the graph. It is obtained after a finite number of iterations using the following fix-point function:

$$\pi^{(k+1)} = \pi^{(k)}G'' \tag{1}$$

To facilitate the computation of the PageRank vector π , Eq. 1 is expressed in the following formula [4]:

$$\pi^{(k+1)} = \pi^{(k)}G'' = \pi^{(k)} [d(G + \delta\omega) + (1 - d)\mathbb{1}v]$$

in which $\mathbb{1}$ is a column vector whose all n entries are equal to 1. This means:

$$\pi^{(k+1)} = d\pi^{(k)}G + d(\pi^{(k)}\delta)\omega + (1 - d)v \tag{2}$$

4 Personalized PageRank for Measuring Semantic Similarity

In the **Linked Open Data** cloud, **RDF** graphs are used to represent semantically connected information resources. For assessing similarity between resources in an **RDF** graph, nodes, links, and their relationships are incorporated into calculation. These are considered as features of the graph and can be utilized in similarity comparison. Features extracted from a resource are discriminative and can be used to distinguish itself from other objects of the same type. Feature-based metrics first attempt to represent a resource as a set of features and then calculate the similarity between two resources by matching their corresponding collections of features. Two resources are considered to be more similar if they share more

common features [1]. As a base for our evaluation, in the preceding sub-sections we review a similarity metric utilizing the PageRank algorithm to characterize a resource in an RDF graph by the rank of the surrounding resources as a feature vector. The similarity between two resources is computed by the cosine similarity of the two feature vectors.

Based on the original PageRank algorithm, the Personalized PageRank algorithm was derived to measure the similarity between topics as proposed in [6]. In this approach, topic is comprised of a set of words, represented as nodes in a graph and PageRank vector is used to characterize a topic. By modifying the personalization vector v in Eq. 2, the corresponding entries to the topic are assigned the value of 1, whilst all the other entries of v are assigned the value of 0. This helps concentrate all probability mass to the constituent words, thereby characterizing the topic. By doing this the topic is biased and earns a high rank [7]. The PageRank vector obtained in this way can be considered as the features of the topic and helps distinguish the topic from others. Eventually, the similarity between two topics r_a and r_b represented by vector $\pi_a = \{a_i\}_{i=1,\dots,n}$ and $\pi_b = \{b_i\}_{i=1,\dots,n}$ is computed as the inner product between the two vectors [2].

$$p.\text{PageRank}(r_a, r_b) = \frac{\pi_a \cdot \pi_b}{\|\pi_a\| \|\pi_b\|} = \frac{\sum_{i=1}^n a_i \times b_i}{\sqrt{\sum_{i=1}^n (a_i)^2} \times \sqrt{\sum_{i=1}^n (b_i)^2}} \quad (3)$$

Personalized PageRank has been successfully applied to measure similarity in WordNet [8, 9]. In this paper, we investigate the possibility of using the metric for computing similarity between resources in an RDF graph. In the preceding section, we are going to perform experiments on a **Linked Data** dataset to analyze the performance of Personalized PageRank.

5 Evaluation

5.1 Experimental Setup

It is highly beneficial to conduct an analysis on the performance of the metric to provide an insight into the suitability and the effectiveness of the measurement technique for **Linked Data**. This helps reveal how effective Personalized PageRank can be in computing similarity, with regard to a ground-truth dataset. To evaluate the efficacy of a similarity metric, it is essential to compare its computational outcomes with a gold standard. A ground-truth dataset or gold standard contains associations where the similarity between artists/bands reflects human perception of the likeness among musical artists. Last.fm is an online music website that specializes in collecting and processing user's musical preferences. Alongside other useful information, Last.fm also provides a list of similar artists/bands for each artist/band. The similarity degree between a pair of artists/bands is represented in two forms: numerical and verbal values. Numerical scores are bound to the interval $[0, 1]$, with the value of 1 corresponds to

absolute similarity while the value of 0 denotes complete difference. The similarity is also expressed verbally in the following degrees: *Super similarity*, *Very high similarity*, *High similarity*, *Medium similarity*. The similarity degrees defined by Last.fm are not symmetric, e.g.: The degree of similarity of The Beatles to Pink Floyd is *High similarity*, whereas the degree of similarity of Pink Floyd to The Beatles is *Very high similarity*.

We re-implement Personalized PageRank and perform experiments to investigate its performance. For comparison, we use a dataset consisting of 1000 artists/bands from Last.fm using the provided API². As the input for the calculation of Personalized PageRank, we retrieve data from a server containing DBpedia 3.8. Features are collected from the server through its SPARQL endpoint by means of Jena Apache³. Starting from the original list, data for every artist/band from DBpedia as well as similar artists/bands is scraped. Since not all similar artists/bands in Last.fm have their counterpart in DBpedia, we discard any artist/band having less than 40 similar items from the list. After this step, 820 artists/bands remain in the final list. The data is saved into local files for further processing. The similarity calculation is performed on the data in the external files.

5.2 Neighborhood Graph

For feature-based similarity metrics, it is necessary to collect a set of features for each resource. In the first place, a set of properties needs to be specified. For DBpedia, we selected 20% most popular properties of the DBpedia ontology used in the musical domain apart from `dbpedia-owl:wikiPageWikiLink`, plus `owl:sameAs`, `rdf:type`⁴ and `dcterms:subject`⁵. The SPARQL query in Listing 1.1 is used to retrieve the list of outgoing properties in the musical domain in descending order according to their frequency of occurrence. The selection of incoming properties can be done in the same manner.

```
SELECT ?p (COUNT(?p) as ?n) WHERE { {
  ?s ?p ?o .
  ?s rdf:type dbpedia-owl:MusicalArtist
} UNION {
  ?s ?p ?o .
  ?s rdf:type dbpedia-owl:Band }
} ORDER BY DESC(?n)
```

Listing 1.1. Retrieving outbound properties

An RDF graph spreads out on numerous resources, consisting of several layers of edges (predicates). For collecting feature sets, it is obviously infeasible

² www.last.fm/api.

³ <http://jena.apache.org/>.

⁴ <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>.

⁵ <http://purl.org/dc/terms/subject>.

to consider all nodes and edges in it. Therefore, we collected a set of features by expanding a graph using the selected set of properties for a limited depth. Considering a pair of resources that are involved in the similarity calculation, a neighborhood graph was built by expanding from each resource using the selected set of properties as shown in Table 1. For each resource, depending on the type of experiments, features can be collected in one or two levels of edges.

Table 1. The set of properties for collecting features.

<i>Outgoing</i>	
<code>rdf:type</code>	<code>dbpedia-owl:associatedAct</code>
<code>owl:sameAs</code>	<code>dbpedia-owl:influenced</code>
<code>dbpedia-owl:instrument</code>	<code>dbpedia-owl:influencedBy</code>
<code>dbpedia-owl:writer</code>	<code>dbpedia-owl:bandMember</code>
<code>dcterms:subject</code>	<code>dbpedia-owl:formerBandMember</code>
<code>dbpedia-owl:associatedBand</code>	<code>dbpedia-owl:currentMember</code>
<code>dbpedia-owl:associatedMusicalArtist</code>	<code>dbpedia-owl:pastMember</code>
<code>dbpedia-owl:background</code>	<code>dbpedia-owl:occupation</code>
<code>dbpedia-owl:genre</code>	<code>dbpedia-owl:birthPlace</code>
<i>Incoming</i>	
<code>dbpedia-owl:previousWork</code>	<code>dbpedia-owl:producer</code>
<code>dbpedia-owl:subsequentWork</code>	<code>dbpedia-owl:artist</code>
<code>dbpedia-owl:knownFor</code>	<code>dbpedia-owl:writer</code>
<code>dbpedia-owl:award</code>	<code>dbpedia-owl:associatedBand</code>
<code>dbpedia-owl:album</code>	<code>dbpedia-owl:associatedMusicalArtist</code>
<code>dbpedia-owl:notableWork</code>	<code>dbpedia-owl:musicalArtist</code>
<code>dbpedia-owl:lastAppearance</code>	<code>dbpedia-owl:musicalBand</code>
<code>dbpedia-owl:basedOn</code>	<code>dbpedia-owl:musicComposer</code>
<code>dbpedia-owl:starring</code>	<code>dbpedia-owl:bandMember</code>
<code>dbpedia-owl:series</code>	<code>dbpedia-owl:formerBandMember</code>
<code>dbpedia-owl:openingFilm</code>	<code>dbpedia-owl:starring</code>
<code>dbpedia-owl:related</code>	<code>dbpedia-owl:composer</code>

Furthermore, also depending on the purpose of measurement, an extension can either be done using only outbound edges or using both inbound and outbound edges. Figure 1 illustrates the process of collecting features. The big nodes with color represent starting resources; small color circles with number correspond to expansion steps. In the figure an undirected edge represents properties in both incoming and outgoing directions.

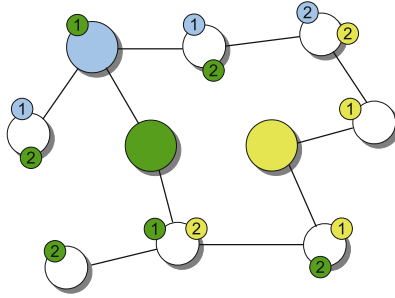


Fig. 1. Collecting feature sets, depth $d = 2$

5.3 Results

In order to investigate the effect of the selection of feature sets on the outcome, we carried out experiments using independent settings. First, we considered different levels of depth and then in each setting, the selection of properties for collecting a set of features. Similarity calculation is performed on the neighborhood graph data. The list of similarity values calculated by Personalized PageRank is then compared with the corresponding ground-truth list from Last.fm using RMSE (Root Mean Squared Error) and Spearman's rank correlation coefficient. The RMSE index is computed as follows:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (4)$$

in which y_i is the value computed by Personalized PageRank; \hat{y}_i is the corresponding value extracted from ground-truth data, n is the number of samples.

We calculate similarity for 1-hop features and 2-hop features. The distribution of the RMSE values for 820 artists/bands is shown in the boxplot diagram in Fig. 2. This figure suggests that for this set of data, PageRank's measurement outcomes are comparable for both cases. This means that the selection of two-hop features does not contribute to a great change in performance.

Furthermore, we investigate how well the ranking produced by Personalized PageRank is associated with the ranking by Last.fm. For this evaluation, the Spearman's rank correlation coefficient metric is utilized to analyze the degree of correlation. For two ranked variables $x = \{X_i\}_{i=1, \dots, n}$ and $y = \{Y_i\}_{i=1, \dots, n}$ the Spearman's rank correlation coefficient ρ is defined as follows:

$$\rho = \frac{\sum_{i=1}^n (X_i - \mu_X) \times (Y_i - \mu_Y)}{\sqrt{\sum_{i=1}^n (X_i - \mu_X)^2 \times \sum_{i=1}^n (Y_i - \mu_Y)^2}} \quad (5)$$

in which μ_X and μ_Y are the mean of X and Y , respectively.

This metric reflects the similarity between the order sorted by Personalized PageRank and that of Last.fm. This is highly important since the ranking of a metric helps identify the most similar artists/bands to a given artist/band.

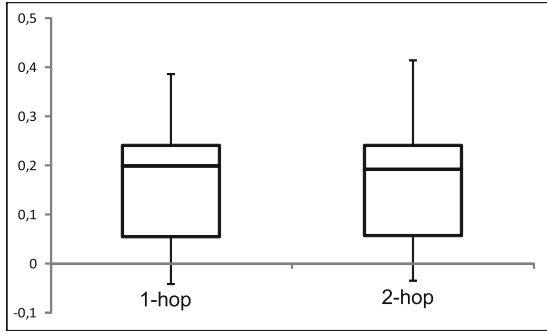


Fig. 2. RMSE

For each artist/band of a dataset, the list of similar artists/bands is ranked according to their similarity scores. This list is then compared with the ranking of the corresponding artist/band of the ground-truth datasets. Figure 3 depicts the histogram of the Spearman’s rank correlation coefficient calculated for the datasets from Last.fm. For the sake of presentation, the histogram values are converted to percentage (%). In this diagram, each bar depicted in label v of the x-axis represents the percentage of artists/bands having the value Spearman’s rank coefficient ρ in the interval $\rho \in [v; v + 0, 1)$. The figure implies that the similarity results produced by Personalized PageRank are partly correlated with the ground-truth data. In addition, we see that also for this measurement metric, the 1-hop features and 2-hop features can help produce similar outcomes.

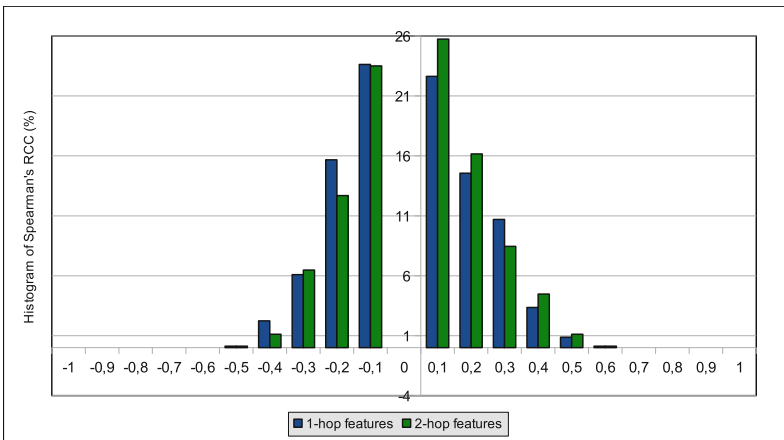


Fig. 3. Spearman’s rank correlation coefficient

6 Conclusions and Future Work

Our paper has introduced an investigation into a metric for measuring similarity based on the structural context of a graph, Personalized PageRank. In our approach, we analyzed the potential of using the metric for automatically measuring similarity in **Linked Data**. The similarity values are considered to be useful since they can be used for recommendation tasks. To validate the outcomes, the results computed by the metric are compared with a ground-truth dataset from Last.fm. Experimental results show that in the given contexts, Personalized PageRank is capable of producing adequate results with regard to the Last.fm dataset. We acknowledge that the selection of properties for expanding a neighborhood graph in **Linked Data** can have an influence on the final outcome. From our perspective, how to opt for the best set of properties in calculating similarity remains an open research question.

References

1. Tversky, A.: Features of similarity. *Psychol. Rev.* **84**(4), 327–352 (1977)
2. Turney, P.D., Pantel, P.: From frequency to meaning: vector space models of semantics. *J. Artif. Intell. Res.* **37**(1), 141–188 (2010)
3. Page, L., Brin, S., Motwani, R., Winograd, T.: The pagerank citation ranking: bringing order to the web. Technical Report, Stanford InfoLab (1999)
4. Wills, R.S.: Google’s PageRank: the math behind the search engine. *J. Math. Intelligencer* **28**(4), 6–10 (2006)
5. Rossi, R.A., Gleich, D.F.: Dynamic PageRank using evolving teleportation. In: Bonato, A., Janssen, J. (eds.) *WAW 2012. LNCS*, vol. 7323, pp. 126–137. Springer, Heidelberg (2012)
6. Haveliwala, T.H.: Topic-sensitive PageRank. In: *Proceedings of the 11th International Conference on World Wide Web (WWW 2002)*, pp. 517–526. ACM (2002)
7. Garla, V.N., Brandt, C.: Semantic similarity in the biomedical domain: an evaluation across knowledge sources. *BMC Bioinformatics* **13**, 1–13 (2012)
8. Agirre, E., Cuadros, M., Rigau, G., Soroa, A.: Exploring knowledge bases for similarity. In: *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC 2010)*, pp. 19–21 (2010)
9. Agirre, E., Soroa, A.: Personalizing PageRank for word sense disambiguation. In: *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2009)*, pp. 33–41 (2009)
10. Zaveri, A., Rula, A., Maurino, A., Pietrobon, R., Lehmann, J., Auer, S.: Quality assessment for linked data: a survey. *Seman. Web J.* (2015)
11. Rula, A., Zaveri, A.: Methodology for assessment of linked data quality. In: *Proceedings of the 1st Workshop on Linked Data Quality, LDQ@SEMANTiCS (2014)*
12. Juran, J.M., Gryna, F.M.: *Juran’s Quality Control Handbook*. McGraw-Hill(Industrial engineering series), New York (1988)
13. Passant, A.: dbrec — Music recommendations using DBpedia. In: Patel-Schneider, P.F., Pan, Y., Hitzler, P., Mika, P., Zhang, L., Pan, J.Z., Horrocks, I., Glimm, B. (eds.) *ISWC 2010, Part II. LNCS*, vol. 6497, pp. 209–224. Springer, Heidelberg (2010)

14. Passant, A.: Measuring semantic distance on linking data and using it for resources recommendations. In: AAAI Spring Symposium: Linked Data Meets Artificial Intelligence (2010)
15. Freitas, A., Oliveira, J.G., O’Riain, S., Curry, E., Pereira da Silva, J.C.: Querying linked data using semantic relatedness: a vocabulary independent approach. In: Muñoz, R., Montoyo, A., Métais, E. (eds.) NLDB 2011. LNCS, vol. 6716, pp. 40–51. Springer, Heidelberg (2011)
16. Freitas, A., Oliveira, J.G., O’Riain, S., Curry, E., Pereira da Silva, J.C.: Treo: best-effort natural language queries over linked data. In: Muñoz, R., Montoyo, A., Métais, E. (eds.) NLDB 2011. LNCS, vol. 6716, pp. 286–289. Springer, Heidelberg (2011)
17. Mendes, P.N., Mühleisen, H., Bizer, C.: Sieve: linked data quality assessment and fusion. In: Proceedings of EDBT-ICDT 2012, pp. 116–123. ACM (2012)
18. Gunawardana, A., Meek, C.: A unified approach to building hybrid recommender systems. In: Proceedings of RecSys 2009, pp. 117–124. ACM (2009)
19. Kontokostas, D., Westphal, P., Auer, S., Hellmann, S., Lehmann, J., Cornelissen, R., Zaveri, A.: Test-driven evaluation of linked data quality. In: Proceedings of WWW 2014, ACM (2014)
20. Schultz, A., Matteini, A., Isele, R., Bizer, C., Becker, C.: LDIF - linked data integration framework. In: Proceedings of COLD 2011 (2011)

Opinion Analysis in Social Networks Using Antonym Concepts on Graphs

Hiram Calvo^(✉)

Centro de Investigación en Computación, Instituto Politécnico Nacional,
Center for Computing Research, National Polytechnic Institute,
Av. Juan de Dios Bátiz e/M.O. Mendizábal s/n, Nva. Ind., 07738
Vallejo, Mexico
hcalvo@cic.ipn.mx

Abstract. In sentiment analysis a text is usually classified as positive, negative or neutral; in this work we propose a method for obtaining the relatedness or similarity that an opinion about a particular subject has with regard to a pair of antonym concepts. In this way, a particular opinion is analyzed in terms of a set of features that can vary depending on the field of interest. With our method, it is possible, for example, to determine the balance of honesty, cleanliness, interestingness, or expensiveness that is expressed in an opinion. We used the standard similarity measures Hirst-St-Onge, Jiang-Conrath and Resnik from WordNet; however, finding that these measures are not well-suitable for working with all Parts-of-Speech, we additionally proposed a new measure based on graphs, to properly handle adjectives. We validated our results with a survey to a sample of 20 individuals, obtaining a precision above 82 % with our method.

Keywords: Sentiment analysis · Opinion mining · Adjective similarity measure · Wordnet · Antonyms

1 Introduction

In the area of Sentiment Analysis there are several works that classify text from several sources using a training corpus. Usually, these works are focused on finding only their polarity, without few of them considering other kinds of analysis.

Pang and Lee [1] classify movie reviews using automatic learning techniques according to their polarity. Considering the words that correspond to a certain polarity, they experiment with Naïve Bayes, Maximum Entropy and Support Vector Machine (SVM) learning methods.

With the recent increase in the use of social networks, several works study opinions in social networks, being Twitter a common example [2–4]. All these works consider particular features of this social network and use automatic learning techniques such as tree kernel or SVM to classify tweets.

In this work, we propose an unsupervised method that, considering opinions from a social network, provides a quantification of relatedness to a specific list of concepts. This list of concepts is composed in turn by pairs of antonym words, allowing the user

to define a set of features beyond simple polarity. Our method is unsupervised because it does not use machine learning on previously tagged examples. It relies on the WordNet structure to find similarities. Additionally, current similarity measures can operate only on a reduced set of Parts-of-Speech (POS), hindering the full exploitation of opinion texts, which can be a problem in short texts such as those found in Twitter. That is why we propose a graph based similarity measure to be able to consider all POS.

In the following sections we present our method (Sect. 2), then we present frequently used similarity measures (Sect. 3), the experiments and comparison with human evaluators (Sect. 4), and finally we draw our conclusions in Sect. 5.

2 Method for Opinion Mining Using Antonym Concepts

We have divided our method in the following main stages: Replacement, text pre-processing, and comparison. See Fig. 1 for a general diagram of our method.

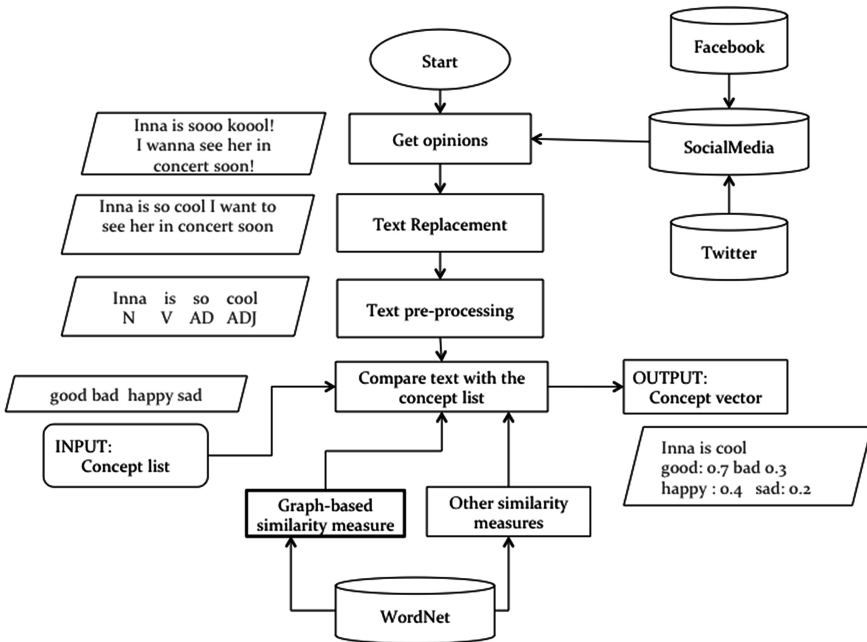


Fig. 1. Diagram of the proposed method for opinion mining.

In our diagram, we have the original input: “Inna is sooo kooool!”; then, after the text replacement, this text becomes “Inna is so cool!”. Afterwards, it is preprocessed and tagged: “inna/N is/V so/AD cool/ADJ”. This text is then compared with a concept

list, consisting in two pairs of antonym concepts: *good/bad*, *happy/sad*. The output consists of a vector that relates each concept with the text: *good: 0.7/bad: 0.3*, *happy: 0.4*, *sad: 0.2*. For obtaining this vector, we use standard similarity measures, and additionally we propose a graph-based similarity measure (shown in bold frame in the diagram). In the following sections we describe each component in detail.

2.1 Replacements

Our replacements database expands contractions, acronyms, emoticons or other ways of quick typing into a normal word representation that can be compared against the concept vector. See Table 1 for an example of replacements.

Table 1. Sample from the *replacements* list.

Acronym, contraction or emoticon	Replacement
aykm	are you kidding me
b4	before
bff	best friend forever
gr8	great
lol	laughing out loud
lv	love
lil	little
sth	something
thx	thank you
arent	aren't
dont	don't
Im	I am
cannot	can not
:), :-), :], :3	happy
XD, :D, = D	laughing
:*	kiss
:O, O.O, o.O, o_o	surprise
:(, :-(, :C, :[sad
:'(crying

2.2 Text Pre-processing

Preprocessing consisted on sentence tokenization, conversion to lowercase, stopwords removal, POS tagging, and negated words identification. For example, the sentence “I didn’t like this movie, but I ...” will become “I didn’t NOT_like NOT_this NOT_-movie, but I”. We also lemmatized the input text.

2.3 Graph Building

We build a graph using WordNet senses and synonyms. To illustrate this, consider the word *angry*. We can see in Table 2 the Synsets of this word, along with its glosses (Fig. 2).

Table 2. Senses (synsets) of *angry*

Sense	Words from synset	Gloss
angry#n#1	angry	<i>feeling or showering anger</i>
angry#n#2	angry, furious, raging, tempestuous, wild	<i>(of elements) as if showing violent anger</i>
angry#n#3	angry	<i>severely inflamed and painful</i>

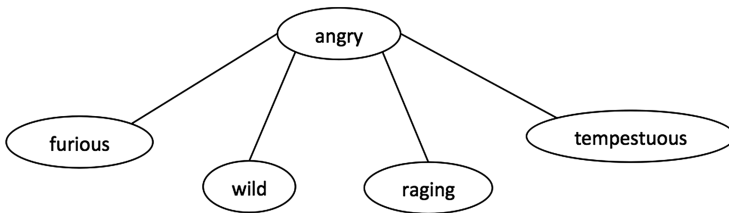


Fig. 2. Subgraph representing the synsets of *angry*

We build the graph of this word, along with its several senses, and then we consulted the synonyms of each one of these words. Once we have expanded one level all of these words, we obtain the following subgraph (Fig. 3).

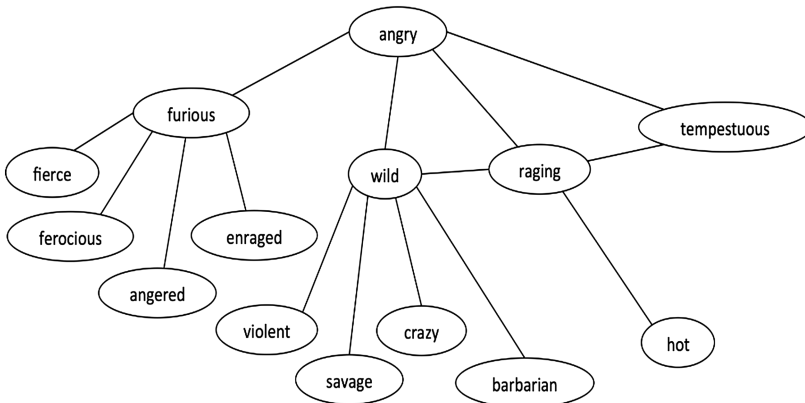


Fig. 3. Synonym subgraph from WordNet after expanding one more level

Subgraphs are created on-demand and are added to the full graph when searching a path between two nodes, as explained on the next section.

2.4 Input Text and Concept List Comparison

Once the graph is built, our goal consists on finding the shortest path between each word from the input text, and each word from the concept list. We implemented the algorithm of bidirectional search along with breadth-first search. With this, we are able to speed up finding a solution. A breadth-first search is started from the root node to the goal node, and other breadth-first search starts from the goal node to the root node. If a concept is in the exploration border of both set, this means both searches have coincided, and the solution will be the union of the both searches: the path from the root node to the goal node through the common concept.

Despite more than one path can be found between two nodes, the expansion method always starts from the first synsets, assuring the path is the shortest one, and that the found concepts are the most relevant possible ones.

We experimented with five variations of this method:

- Edge count
- Use of logarithms
- Logarithms with POS change penalization
- Semantic orientation of words
- Individual semantic orientation.

We describe each one of them in the following sections.

2.4.1 Edge Count

The first variation of the method consists only in using the path between two words in the graph. For each pair of nodes connected, a value of 1 is assigned. The distance of the path is calculated by adding all edges between words. This is repeated for each word from a total of n words from the opinion text t .

For a set that represents the distances of a concept c from the concept list with the words of an opinion $w_i \dots w_n$, we calculate the average of distances d_{avg} as:

$$dp = \frac{\sum_i^n d(w_i c)}{n}$$

Then we divide d_{avg} by the maximum distance dm in the WordNet graph (calculated previously, and equal to 15), so that we can find the relatedness of a concept c with a text t as:

$$dp = \frac{\sum_i^n d(w_i c)}{n}$$

2.4.2 Use of Logarithms

This variation consists on using a different value for the distance between each pair of nodes. The edge between each node will have a value of 0 (no distance) for the first synset of a word, and then it will be increased to 1 using the logarithm function.

Table 3. Synsets of the word *like*

Sense	Words in the same synset	Gloss
like#n#1	like, the-like, the-like-of	– <i>A similar kind</i>
like#n#2	like, ilk	– <i>A kind of person</i>
like#a#1	like, similar	– <i>Resembling or similar; having the same or some of the same characteristics; often used in combination</i>
like#a#2	like, same	– <i>Equal in amount or value</i>
like#a#3	like, alike, similar	– <i>Having the same of similar characteristics</i>
like#a#4	like, comparable, corresponding	– <i>Conforming in every respect</i>
like#v#1	like, wish, care	– <i>Prefer or wish to do something</i>
like#v#2	like	– <i>Find enjoyable or agreeable</i>
like#v#3	like	– <i>Be found of</i>
like#v#4	like	– <i>Feel about or towards; consider, evaluate or regard</i>
like#v#5	like	– <i>Want to have</i>

This is motivated by the fact that the different senses of a word are listed from the most frequent one to the less frequent. For example, see Table 3 for the word *like*.

To assign a distance to each synset of a word, we use the following procedure. First, if a word has more than one synset, let N be the total number of synsets. C is defined as:

$$C = \frac{N - 1}{N + 1}$$

and the distance d_n is defined as:

$$d_n = \log_N(1 + nC)$$

where n is a number that represents the rank of the sense from 1 to N for a given word. If the word corresponding to a synset is shared by all the other synsets in the same POS, we consider N equal to 1, and the distance d_n for that synset is 0. See Table 4 for an example.

Figure 4 shows the resulting graph once each node related with a word is added.

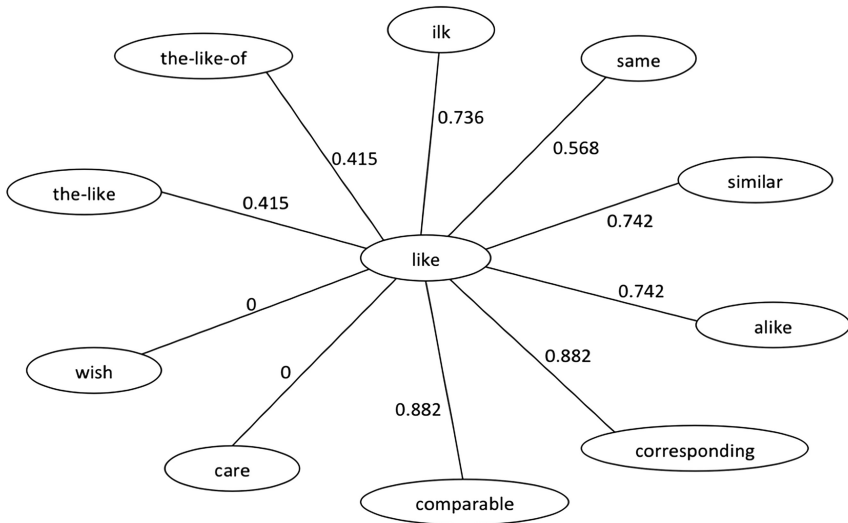
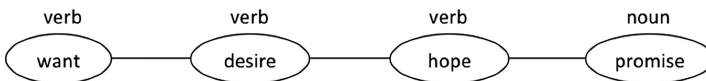
2.4.3 Logarithms with POS Change Penalization

In this variation we consider logarithms as well, but we will take into account the POS when following a path of nodes. Consider Fig. 5.

We can see that we can get from *want* to *promise* through a chain of verbs, but also allowing a POS change from hope to promise. We will add 1 to the total distance of a path for each POS change.

Table 4. Distances for each synset of *like*

Sense	Words in the same synset	Distance
like#n#1	like, the-like, the-like-of	$d_n = \log_2 (1 + (1*1/3)) = 0.415$
like#n#2	like, ilk	$d_n = \log_2 (1 + (1*1/3)) = 0.736$
like#a#1	like, similar	$d_n = \log_4 (1 + (1*3/5)) = 0.339$
like#a#2	like, same	$d_n = \log_4 (1 + (2*3/5)) = 0.568$
like#a#3	like, alike, similar	$d_n = \log_4 (1 + (3*3/5)) = 0.742$
like#a#4	like, comparable, corresponding	$d_n = \log_4 (1 + (4*3/5)) = 0.882$
like#v#1	like, wish, care	$d_n = 0$
like#v#2	like	$d_n = 0$
like#v#3	like	$d_n = 0$
like#v#4	like	$d_n = 0$
like#v#5	like	$d_n = 0$

**Fig. 4.** Adding and weighting related nodes**Fig. 5.** Path from *want* to *promise*

2.4.4 Semantic Orientation of Words

In the previous variations, in order to calculate the relationship a text has with a list of concepts, each one of the concepts was considered separately. Let us remember that our

motivation is that the list of concepts is conformed not by unrelated concepts, but by pairs of antonym concepts. In this variation we consider the distance between each member of these pairs as well, integrating it in a single formula.

Peng et al. [5] measure the semantic orientation of words considering three factors. In this variation we propose measuring in a similar manner the distance of words conforming the text, with the pairs of antonyms from the list of concepts. The used formula is:

$$OS(w) = \frac{dis(w, c_1) - dis(w, c_2)}{dis(c_1, c_2)}$$

We measure the distance of each one of the words w from the text with the pair of antonyms (c_1 y c_2) at the same time. This formula returns a numeric result between -1 and 1 . If the result is less than 0 , we can say that w is closer to c_1 , and if the result is greater than 0 , w is closer to c_2 . This formula is applied to all words conforming the text, and with all each pair of antonyms, and the result is their average.

2.4.5 Individual Semantic Orientation

The fifth and last variation of this method is based on the previously described variation. In this variation, distances to each element of the pair of antonym concepts are calculated separately, that is:

$$OSi(w) = 1 - \frac{dis(w, c_1)}{dis(c_1, c_2)} \quad OSi(w) = 1 - \frac{dis(w, c_2)}{dis(c_1, c_2)}$$

When any of these measures is less than 0 , the semantic orientation is set to 0 .

As in the previous method, the average of all words for each concept represents the relatedness of the text with each of them.

3 Similarity Measures

In order to compare our proposed graph-based similarity measure, we compare each concept of the antonym pairs concept list using standard WordNet measures, such as Hirst-St-Onge, Jiang-Conrath, and Resnik. The latter two measures are not able to handle adjectives, but we include them in our experiments because they are frequently used similarity measures [6].

3.1 Hirst-St-Onge

This similarity measure was proposed in 1998 [7]; it quantifies semantic relationships between words and it is path-based. This measure establishes the relation between two concepts trying to find a path between them. This path is sought to be not too long and trying to avoid frequent changes in its direction. From a node, it traverses all relationships horizontally, up or downwards, and penalizes changes of direction. As an example, the is-a relationships are upwards, while the has-part ones are horizontal.

3.2 Jiang-Conrath

The similarity measure proposed by Jiang and Conrath [8] uses the concept of Information Content, defined as the negative of the logarithm of the probability of a certain concept:

$$IC(c) = -\log P(c)$$

The Jiang-Conrath similarity is defined then as:

$$\text{sim}_{JCN}(c_1, c_2) = \frac{1}{IC(c_1) + IC(c_2) - 2 * IC(LCS(c_1, c_2))}$$

There are two special cases that we should avoid when calculating similarity with this method: both involve the case when the denominator is zero. The first case occurs when $IC(c_1)$, $IC(c_2)$ and $IC(LCS(c_1, c_2))$ are zero. In most cases, this only happens when the three concepts are the root node, however, when a concept has a frequency of zero, IC turns to be zero as well. The second case occurs when the concepts c_1 and c_2 are the same. Intuitively, this would mean maximum similarity, and the result would be infinite. Because this is impossible, the smallest distance possible is used, and its inverse is returned.

3.3 Resnik

Resnik [9] defines $P(c)$ as the probability that selecting a random word it is an instance of the concept c . Formally, there is a variable different to random, which runs on each word associated to each concept on the hierarchy. Given a concept, each observed noun is a member of this concept with a probability $P(c)$ or it is not a member, with a probability of $1 - P(c)$. Because each word is a member of the root node, called *entity* in WordNet, it means that the probability of *root* is one, and the probability of the nodes below is lower, decreasing as deepness increases.

Given a hierarchy of hyponyms and a corpus, to obtain $P(c)$ the frequency of each concept c is counted, as well as the frequency of its parents up to the root concept in the hierarchy. $words(c)$ is defined as the set of words that have the node c as ancestor, including c itself. With this, we can calculate the probability of the concept c as:

$$P(c) = \frac{\sum_{w \in words(c)} count(w)}{N}$$

This is the probability of the concept c . The probability of a random word is an instance of this concept. Once all probabilities are calculated for each node, we can add them to the hierarchy. With such probabilities, we can then compute the amount of Information Content of a concept c as:

$$IC(c) = -\log P(c)$$

We define also the immediate upper node that includes c_1 and c_2 in the hierarchy as $LCS(c_1, c_2)$, meaning *least common subsumer*.

Resnik argues that the similarity between two words is related to how much information they share in common, so that, the more information they have in common, more similar they are. If we have two concepts, their commonality is defined by what they inherit from their common ancestor, *i.e.*, the lowest node in the hierarchy that includes both. This is expressed by the formula:

$$\text{sim}_{\text{resnik}}(c_1, c_2) = \text{IC}(LCS(c_1, c_2))$$

4 Experiments and Results

In this section we present our evaluation method, consisting on a survey to several individuals, and then we present the results of our method, comparing all the proposed variants of the graph-based similarity measure, and standard similarity measures.

4.1 Evaluation Method

We provided a set of opinions to 20 individuals, along with a list of concepts, asking them to select one concept from each pair of antonym concepts, that they would believe it described the opinion.

The set of opinions ranged from 5 to 7 for each product or person, as listed below (Table 5).

Table 5. Opinion list and concept list

Opinions	Product/person	Concept list
6	Videogame console	cheap/expensive
6	Electronics brand	good/bad, pretty/ugly, easy/hard
6	Football player	good/bad, proud/modest, skillful/unskillful
5	President	good/bad, honest/corrupt
5	Videogame	good/bad, entertaining/boring
7	Soap	good/bad, clean/dirty
5	City	good/bad, cold/hot, pretty/ugly

For example, for the following opinion the participants should choose one of each pair of antonyms:

Easy to set up, the graphics are outstanding. Easy to use the interface. The controller is much better than the PS3 design. The PSN network is much more affordable than the ‘other box’ and more cheaper. I do not regret this purchase. Just waiting for more awesome games!

good _____ bad _____ cheap _____ costly _____

That is, they should tick one of (good or bad) and one of (cheap or costly).

4.2 Results

We compared the results of our 5 proposed variants, along with standard similarity measures in WordNet, namely Hirst-St-Onge, Jiang_Conrath, and Resnik. The two latter measures do not work for adjectives. Additionally, we provide a comparison of the kind of relationships used for calculating the distance: Using all relationships vs. using only adjectives. We can see that using all relationships helps noticeably in most cases.

Table 6. Evaluation of each variant and similarity measures.

Method	Relationships	Recall	Precision
Edges	All	74.22 %	75.0 %
	Adjectives	64.94 %	70.78 %
Logarithms	All	63.91 %	63.91 %
	Adjectives	67.01 %	67.70 %
Log with POS change penalization	All	53.60 %	54.16 %
	Adjectives	54.63 %	54.63 %
Semantic Orientation (SO)	All	74.22 %	75.78 %
	Adjectives	63.91 %	67.39 %
Individual SO	All	80.41 %	82.10 %
	Adjectives	59.79 %	67.44 %
Hirst-St-Onge	All	28.86 %	68.29 %
Jiang-Conrath	Nouns only	52.57 %	53.68 %
Resnik	Nouns only	13.40 %	61.90 %

5 Conclusions and Future Work

WordNet is a highly connected graph containing many definitions and senses that are not always frequently used. This implies that sometimes words that one would expect to be far are closely connected. For example, *good* and *bad* are connected by a distance of only 4 edges. This causes that in many cases the similarity of an opinion with a concept in the list is very close to its similarity with its antonym.

One of the main limitations for this work is that all words must be connected to the graph. Surprisingly, some words, such as *beautiful* have no synonymy relationship with other words. One could expect *beautiful* to be synonyms with *pretty*, or at least to find a short path between them. This leads to the conclusion that one must be careful when selecting the list of antonym concepts.

The JCN and Resnik measures were not very helpful for our purposes, since they are not able to handle adjectives. The Hirst-St-Onge measure is able to provide a similarity for all four POS, but we can see from Table 6 that its performance is lower, compared to our proposed method. At the present moment, we are not aware of other similarity measure on WordNet that considers all four POS yielding a better performance.

The best variation of our method was Individual Semantic orientation. This is because it considers the distance of the antonym concepts as a normalization factor, keeping only those that have a distance lesser than the distance between the antonym concepts, and discarding those that are more distant.

Finally, we can conclude that using distances on the graph of WordNet, created by its synonymy relationships, is a viable way for classifying texts in a variety of groups according to determined features, extending the possibilities beyond classifying them only by their polarity.

As a future work, we plan to explore creating the graph from WordNet from other relationships other than synonymy, and experimenting with other features from the text, aside from paths or distance between concepts.

References

1. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up? Sentiment classification using machine learning techniques. In: EMNLP-2002, pp. 79–86 (2002)
2. Birmingham, A., Smeaton, A.: Classifying sentiment in microblogs: is brevity and advantage? In: ACM International Conference on Information and Knowledge Management, Toronto, Ontario, Canada (2010)
3. Agarwal, A., Xie, B., Vovsha, I., Rambow, O., Passonneau, R.: Sentiment analysis of Twitter data. In: Proceedings of ACL Workshop on Languages in Social Media, pp. 30–38 (2011)
4. Kouloumpis, E., Wilson, T., Moore, J.: Twitter sentiment analysis: the good the bad and the omg. In: Proceedings of the ICWSM 2011 (2011)
5. Peng, Q., Zhao, L., Yu, Y., Fang, W.: A new measure of word semantic similarity based on WordNet hierarchy and DAG theory. In: International Conference on Web Information Systems and Mining, Shanghai, China (2009)
6. McCarthy, D., Koeling, R., Weeds, J., Carroll, J.: Unsupervised acquisition of predominant word senses. *Comput. Linguist.* **33**(4), 553–590 (2007)
7. Hirst, G., St-Onge, D.: Lexical chains as representation of context for the detection and correction of malapropisms. In: Fellbaum, C. (ed.) *WordNet: An Electronic Lexical Database*, Chap. 13, pp. 305–332. The MIT Press, Cambridge (1998)
8. Jiang, J.J., Conrath, D.W.: Semantic similarity based on corpus statistics and lexical taxonomy. In: Proceedings of International Conference on Research in Computational Linguistics, Taiwan, pp. 19–33 (1997)
9. Resnik, P.: Using information content to evaluate semantic similarity in a taxonomy. In: Proceedings of the 14th International Joint Conference on Artificial Intelligence, Montréal, Québec (1995)

Sensor Databases and Applications in Smart Home and City

Traffic Speed Data Investigation with Hierarchical Modeling

Tomonari Masada¹(✉) and Atsuhiko Takasu²

¹ Nagasaki University, 1-14 Bunkyo-machi, Nagasaki 8528521, Japan
masada@nagasaki-u.ac.jp

² National Institute of Informatics,
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 1018430, Japan
takasu@nii.ac.jp

Abstract. This paper presents a novel topic model for traffic speed analysis in the urban environment. Our topic model is special in that the parameters for encoding the following two domain-specific aspects of traffic speeds are introduced. First, traffic speeds are measured by the sensors each having a fixed location. Therefore, it is likely that similar measurements will be given by the sensors located close to each other. Second, traffic speeds show a 24-hour periodicity. Therefore, it is likely that similar measurements will be given at the same time point on different days. We model these two aspects with Gaussian process priors and make topic probabilities location- and time-dependent. In this manner, our model utilizes the metadata of the traffic speed data. We offer a slice sampling to achieve less approximation than variational Bayesian inferences. We present an experimental result where we use the traffic speed data provided by New York City.

1 Introduction

In recent years, many text mining applications, which have been mainly addressed by topic modeling approaches, are being addressed by deep learning approaches [20, 22, 23] effectively. However, topic modeling approaches, and more broadly Bayesian approaches, have an advantage that we can *explicitly* incorporate our domain-specific knowledge in the form of prior distributions when constructing probabilistic models in an application-dependent manner [3, 7, 8, 13]. In this paper, we propose a new topic model for traffic speed analysis and use Gaussian process priors [18] to incorporate our domain-specific knowledge on traffic speeds measured in the urban environment. Since we adopt Bayesian hierarchical modeling for investigating traffic speed data, we call our method *TRINH* (TRaffic speed data INvestigation with Hierarchical modeling).

The aim of our research is to analyze traffic speed data for extracting typical patterns of traffic speed distribution. Traffic speed data are obtained based on vehicles' traveling times given by sensors placed along road segments. We practically analyze traffic speed data to foresee heavy traffic congestion or to find bottlenecks in redesigning traffic light timings [17].

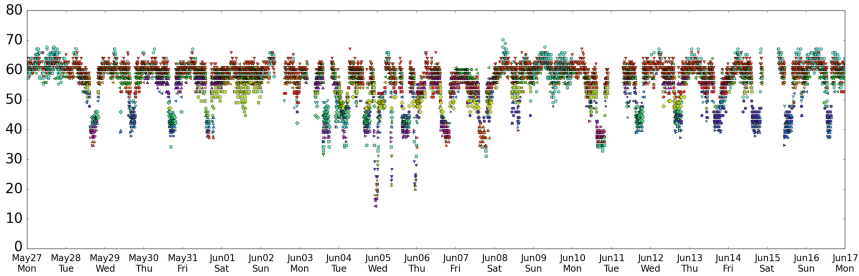


Fig. 1. An example of the analysis achieved by TRINH. We present the data measured by the sensor #289 [1]. The horizontal axis is the time axis ranging from May 27 to June 16 in 2013. The vertical axis shows traffic speeds in mph. The traffic speeds represented by the marker of the same shape and color are analyzed by TRINH as coming from the same gamma distribution. In this case, the number of gamma distributions is 30.

However, traffic speeds show a wide variety of distributions depending on the hours of the day or days of the week. Therefore, for any segment of the sequence of the traffic speeds measured by an individual sensor, it is difficult to tell whether the traffic speed measurements contained in the segment are generated from the same probability distribution or from a mixture of different probability distributions. If the length of the segment is long, e.g. 24 hours, it is better to model the measurements contained in the segment with a mixture of multiple probability distributions. However, if the length is short, e.g. 10 min, it is better to model the measurements with a single distribution.

A similar problem arises in text mining. In many text mining applications, especially in topic extraction or in text summarization, it is difficult to tell whether some sequence of words is generated from the same word distribution or from a mixture of different word distributions. To address this problem, many existing works adopt topic modeling approaches like latent Dirichlet allocation (LDA) [5]. In topic modeling, we can regard each document as a composite of parts each corresponding to different topics and thus can model each document with a mixture of multiple word probability distributions. Our traffic speed analysis also views traffic speed sequences obtained within windows of fixed length as a mixture of diverse traffic speed distributions (cf. Fig. 1).

Traffic speeds are, however, non-negative real numbers. Therefore, we replace multinomial distribution in LDA with gamma distribution. While truncated Gaussian distribution may also be used, we prefer the simplicity of gamma distribution. Consequently, our probabilistic model generates each traffic speed sequence as follows in an LDA-like manner. For each traffic speed sequence regarded as a document, we draw a topic from the per-document topic multinomial distribution and then draw a traffic speed from the gamma distribution that corresponds to the drawn topic. Further, TRINH has two features as follows.

While a generative model obtained in this manner has already been reported [12], the authors fail to capture the domain-specific aspects. Traffic speeds are measured by the sensors each having a fixed physical position.

Further, the measurement is performed in a real-time manner. Therefore, traffic speeds are *location- and time-dependent*. We incorporate this domain-specific knowledge into our model by using prior distributions. This is the first special feature of TRINH. To be precise, we propose a probabilistic model using two Gaussian process priors [18] for modeling location- and time-dependency of traffic speeds. The functions drawn from the one Gaussian process prior are evaluated at the domain points each corresponding to sensor locations, and those drawn from the other Gaussian process prior are evaluated at the domain points each corresponding to measurement time points. The probability of each topic is obtained by applying the logistic function to the sum of the factors including these location- and time-dependent factors. In this manner, we can make topic probabilities dependent on the locations and the time points of traffic speed measurements. In short, our model can utilize the *metadata* of the traffic speed data.

When we define probabilities based on several different factors, the logistic function is often used [6, 9, 15]. However, the posterior inference of the existing approaches is mainly performed by maximizing a variational lower bound of the evidence. Note that this type of inference, often called variational Bayesian inference, introduces an approximation. In this paper, we propose a sampling-based inference. To be precise, we adopt the *general slice sampler* [19, p. 326], which can be used for obtaining samples from the distribution whose density is proportional to the product of positive functions. This technique is efficient when the inverse of each positive function has a simple form. Owing to the general slice sampler, a large part of our inference can be described as sampling from a uniform distribution. Consequently, its implementation is not that complicated. This is the second special feature of TRINH. Our inference is similar to that proposed in [14], because this is also an application of the general slice sampler.

We evaluate TRINH in terms of test data log likelihood. Based on the posterior distribution obtained by our inference, the log likelihood of the test data can be computed. In the experiment, the test data set is prepared as a randomly selected 20 percent of the given data set. The result shows that our method is slightly inferior to the probabilistic model that uses no location and time-stamp data with respect to the best log likelihood achieved during hundreds of iterations of the inference. However, the result also shows that our method can increase the log likelihood more rapidly than the compared model.

The rest of the paper is organized as follows. Section 2 describes the details of TRINH. Section 3 gives a sampling-based inference. Section 4 explains the settings and the results of the experiment. Section 5 reviews some existing proposals. Section 6 concludes the paper with an outlook on future work.

2 Proposal

Our model TRINH is proposed based on latent Dirichlet allocation (LDA) [5, 10]. In this paper, each sequence of the traffic speeds measured by the same sensor within 24 hours from 0 a.m. is regarded as a document. That is, all sensors

produce one document per day. We denote the total number of documents by D and denote the number of traffic speeds contained in the d th document by N_d , which corresponds to the document length in case of text data. Traffic speeds are non-negative real numbers. Therefore, we replace per-topic word multinomial distributions in LDA with gamma distributions. We denote the number of topics by K . The gamma distribution for the k th topic is denoted by Gamma(α_k, β_k), where α_k and β_k are the shape and rate parameters. Our aim is to extract diverse patterns of traffic speed distributions as K different gamma distributions.

TRINH generates traffic speed data in an LDA-like manner as follows. We first draw a topic from the per-document topic multinomial distribution and then draw a traffic speed from the gamma distribution that corresponds to the drawn topic. However, traffic speeds have domain-specific features. Therefore, we define the topic probability by combining the three parameters m_{dk} , λ_{ks_d} , and τ_{kt} as $\theta_{dtk} \propto \exp(m_{dk} + \lambda_{ks_d} + \tau_{kt})$, where s_d is the index of the sensor that measures the traffic speeds contained in the d th document, and t represents a time point at which the measurement occurs. That is, topic probabilities depend on by which sensor the traffic speeds are measured and also on at which time point they are measured. We explain the three parameters below.

The parameter m_{dk} only represents the fact that the probability of the k th topic is different for each document. That is, m_{dk} reflects the basic idea of LDA that different documents have different mixing proportions of topics. However, the probabilities of the same topic for different documents may show some similarity. Therefore, we draw m_{1k}, \dots, m_{Dk} from the Gaussian prior distribution $\mathcal{N}(\mu_k, \sigma_k)$, where μ_k and σ_k are the mean and standard deviation parameters. This prior is used in place of the Dirichlet prior in LDA. The parameter λ_{ks} represents the feature that topic probabilities depend on sensors in such a manner that the same sensor s tends to give similar traffic speed distributions even on different days. The parameter τ_{kt} represents the feature that topic probabilities depend on measurement time points in such a manner that similar traffic speed distributions may be obtained at the same time point of the day even from different sensors. By combining these three parameters, we define the probability of the k th topic at the time point t in the d th document as $\theta_{dtk} \equiv \exp(m_{dk} + \lambda_{ks_d} + \tau_{kt}) / \sum_{k'} \exp(m_{dk'} + \lambda_{k's_d} + \tau_{k't})$.

Additionally, TRINH addresses the two types of similarity, i.e., the similarity among sensor locations and the similarity among time points, by applying Gaussian process priors [18] to the λ_{ks} s and the τ_{kt} s. First, when two sensors are located close to each other, they may give similar traffic speed distributions. In other words, λ_{ks} and $\lambda_{ks'}$ may have similar values when the s th and the s' th sensors are located close to each other. Therefore, we draw a random function f_k from the Gaussian process prior GP($\mathbf{0}, \mathbf{K}_S$) for each k , and evaluate the drawn function at the sensor locations. In the experiment, we use the longitude and latitude of the physical locations of the sensors and compute distances between them. Second, we may observe that traffic speeds measured at the time points close to each other may show similar distributions. In other words, τ_{kt} and $\tau_{k't'}$ may have similar values when the t -th and the t' -th time points are close to

each other on the time axis. Therefore, we draw a random function g_k from the Gaussian process prior $\text{GP}(\mathbf{0}, \mathbf{K}_T)$ for each k and evaluate the drawn function at the measurement time points. The covariance functions \mathbf{K}_S and \mathbf{K}_T reflects the similarity among sensor locations and the similarity among time points, respectively. We set the mean function to $\mathbf{0}$ for both priors so that λ_{ks_d} and τ_{kt} represent deviations from m_{dk} . We give the generative description of TRINH:

1. For each topic $k \in \{1, \dots, K\}$,
 - (a) Draw the rate parameter β_k of the gamma distribution $\text{Gamma}(\alpha_k, \beta_k)$ from the global Gamma prior distribution $\text{Gamma}(a, b)$.
 - (b) Draw a function f_k from $\text{GP}(\mathbf{0}, \mathbf{K}_S)$ and set $\lambda_{ks} = f_k(\mathbf{r}_s)$ for each sensor s , where \mathbf{r}_s is the physical location of the s th sensor.
 - (c) Draw a function g_k from $\text{GP}(\mathbf{0}, \mathbf{K}_T)$ and set $\tau_{kt} = g_k(t)$ for each time point t of the day.
2. For each document $d \in \{1, \dots, D\}$,
 - (a) Draw m_{dk} from the Gaussian prior distribution $\mathcal{N}(\mu_k, \sigma_k)$ and set $\theta_{dtk} \equiv \frac{\exp(m_{dk} + \lambda_{ks_d} + \tau_{kt})}{\sum_{k'} \exp(m_{dk'} + \lambda_{k's_d} + \tau_{k't_d})}$, where s_d refers to the sensor that measures the traffic speeds of the d th document.
 - (b) For each $i \in \{1, \dots, N_d\}$,
 - i. Draw a topic z_{di} from the multinomial distribution $\text{Discrete}(\boldsymbol{\theta}_{dt_{di}})$, where t_{di} is the time point at which the i th traffic speed in the d th document is measured.
 - ii. Draw a traffic speed x_{di} from the gamma distribution $\text{Gamma}(\alpha_{z_{di}}, \beta_{z_{di}})$.

We assume that the covariance function of $\text{GP}(\mathbf{0}, \mathbf{K}_S)$ has the following form: $k_L(\mathbf{r}_s, \mathbf{r}_{s'}) \equiv \sigma_{S0}^2 \delta(s = s') + \sigma_{S1}^2 \exp(-|\mathbf{r}_s - \mathbf{r}_{s'}|^2 / 2l_S^2)$, where $\delta(\cdot)$ is equal to 1 if the condition in parentheses is satisfied and to 0 otherwise. Further, we assume that the covariance function of $\text{GP}(\mathbf{0}, \mathbf{K}_T)$ has the following form: $k_T(t, t') \equiv \sigma_{T0}^2 \delta(t = t') + \sigma_{T1}^2 \exp(-(t - t')^2 / 2l_T^2)$.

Consequently, we obtain the full joint distribution of TRINH as below.

$$\begin{aligned}
 & p(\mathbf{x}, \mathbf{z}, \boldsymbol{\beta}, \mathbf{m}, \boldsymbol{\lambda}, \boldsymbol{\tau} | \boldsymbol{\mu}, \boldsymbol{\sigma}, \mathbf{K}_S, \mathbf{K}_T, \boldsymbol{\alpha}, a, b) \\
 &= \prod_{k=1}^K \left[\frac{b^a}{\Gamma(a)} \frac{\beta_k^{a-1+N_k} \alpha_k Y_k^{\alpha_k-1} e^{-\beta_k(b+X_k)}}{\Gamma(\alpha_k)^{N_k}} \prod_{d=1}^D \frac{1}{\sqrt{2\pi\sigma_k^2}} \exp\left\{-\frac{(m_{dk} - \mu_k)^2}{2\sigma_k^2}\right\} \right] \\
 & \cdot \prod_{k=1}^K \frac{1}{\sqrt{(2\pi)^S |\mathbf{K}_S|}} \exp\left(-\frac{\boldsymbol{\lambda}_k^T \mathbf{K}_S^{-1} \boldsymbol{\lambda}_k}{2}\right) \cdot \prod_{k=1}^K \frac{1}{\sqrt{(2\pi)^T |\mathbf{K}_T|}} \exp\left(-\frac{\boldsymbol{\tau}_k^T \mathbf{K}_T^{-1} \boldsymbol{\tau}_k}{2}\right) \\
 & \cdot \prod_{d=1}^D \prod_{i=1}^{N_d} \prod_{k=1}^K \left\{ \frac{\exp(m_{dk} + \lambda_{ks_d} + \tau_{kt_{di}})}{\sum_{k'} \exp(m_{dk'} + \lambda_{k's_d} + \tau_{k't_{di}})} \right\}^{\delta(z_{di}=k)}, \tag{1}
 \end{aligned}$$

where the following notations are introduced: $N_k \equiv \sum_{d=1}^D \sum_{i=1}^{N_d} \delta(z_{di} = k)$, $X_k \equiv \sum_{d=1}^D \sum_{i=1}^{N_d} \delta(z_{di} = k) x_{di}$, and $Y_k \equiv \prod_{d=1}^D \prod_{i=1}^{N_d} x_{di}^{\delta(z_{di}=k)}$. $\Gamma(\cdot)$ denotes the Gamma function. We consider our sampling method based on the marginalized joint distribution $p(\mathbf{x}, \mathbf{z}, \mathbf{m}, \boldsymbol{\lambda}, \boldsymbol{\tau})$ obtained by integrating out $\boldsymbol{\beta}$ in Eq. (1).

3 Inference

We offer a sampling-based inference for TRINH to achieve less approximation. First, we consider topic assignments \mathbf{z} . We assume that \mathbf{m} , $\boldsymbol{\lambda}$, and $\boldsymbol{\tau}$ are fixed. Then the joint distribution of \mathbf{x} and \mathbf{z} is obtained as follows:

$$p(\mathbf{x}, \mathbf{z} | \mathbf{m}, \boldsymbol{\lambda}, \boldsymbol{\tau}, \boldsymbol{\alpha}, a, b) \propto \prod_{k=1}^K \frac{\Gamma(a + N_k \alpha_k) Y_k^{\alpha_k - 1}}{\Gamma(\alpha_k)^{N_k} (b + X_k)^{a + N_k \alpha_k}} \cdot \prod_{d=1}^D \prod_{i=1}^{N_d} \prod_{k=1}^K \theta_{dt_{di}k}^{\delta(z_{di}=k)}. \quad (2)$$

We assume that $z_{di} = \hat{k}$ before updating z_{di} . Then, based on Eq. (2), the probability that z_{di} is updated to k s.t. $k \neq \hat{k}$ is:

$$p(z_{di} = k | \mathbf{x}, \mathbf{z}^{-di}, \mathbf{m}, \boldsymbol{\lambda}, \boldsymbol{\tau}, \boldsymbol{\alpha}, a, b) \propto \frac{\theta_{dt_{di}k} \Gamma(a + (N_k + 1)\alpha_k)}{x_{di} \Gamma(\alpha_k) \Gamma(a + N_k \alpha_k)} \left(\frac{b + X_k}{b + X_k + x_{di}} \right)^{a + N_k \alpha_k} \left(\frac{x_{di}}{b + X_k + x_{di}} \right)^{\alpha_k}. \quad (3)$$

On the other hand, the probability that z_{di} is updated to \hat{k} is:

$$p(z_{di} = k | \mathbf{x}, \mathbf{z}^{-di}, \mathbf{m}, \boldsymbol{\lambda}, \boldsymbol{\tau}, \boldsymbol{\alpha}, a, b) \propto \frac{\theta_{dt_{di}k} \Gamma(a + N_k \alpha_k)}{x_{di} \Gamma(\alpha_k) \Gamma(a + (N_k - 1)\alpha_k)} \left(\frac{b + X_k - x_{di}}{b + X_k} \right)^{a + N_k \alpha_k} \left(\frac{x_{di}}{b + X_k - x_{di}} \right)^{\alpha_k}. \quad (4)$$

Second, let's consider the sampling of the m_{dk} s. By assuming that \mathbf{z} , $\boldsymbol{\lambda}$, and $\boldsymbol{\tau}$ are fixed, we obtain the full conditional distribution of m_{dk} as follows:

$$p(m_{dk} | \mathbf{m}^{-dk}, \mathbf{x}, \mathbf{z}, \boldsymbol{\lambda}, \boldsymbol{\tau}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) \propto \frac{1}{\sigma_k} \exp \left\{ -\frac{(m_{dk} - \mu_k)^2}{2\sigma_k^2} \right\} \prod_i \frac{\exp(m_{dz_{di}} + \lambda_{z_{di}s_d} + \tau_{z_{di}t_{di}})}{\sum_k \exp(m_{dk} + \lambda_{ks_d} + \tau_{kt_{di}})}, \quad (5)$$

We sample from this posterior by adopting the general slice sampler (cf. Chapter 8.2 [19]). We denote uniform distributions by \mathcal{U} . For each pair of d and k ,

- Draw auxiliary variables u_{dk} and $u_{dk1}, \dots, u_{dkN_d}$ as follows:
 - Draw u_{dk} from $\mathcal{U} \left(\left\{ u : 0 \leq u \leq \frac{1}{\sigma_k} \exp \left\{ -\frac{(m_{dk} - \mu_k)^2}{2\sigma_k^2} \right\} \right\} \right)$.
 - Draw u_{dki} from $\mathcal{U} \left(\left\{ u : 0 \leq u \leq \frac{\exp(m_{dz_{di}} + \lambda_{z_{di}s_d} + \tau_{z_{di}t_{di}})}{\sum_k \exp(m_{dk} + \lambda_{ks_d} + \tau_{kt_{di}})} \right\} \right)$ for each $i = 1, \dots, N_d$.
- Draw m_{dk} from $\mathcal{U}(I_{dk} \cap \bigcap_i I_{dki})$, where

$$I_{dk} \equiv \left\{ m : \mu_k - \sqrt{-2\sigma_k^2 \ln(\sigma_k u_{dk})} \leq m \leq \mu_k + \sqrt{-2\sigma_k^2 \ln(\sigma_k u_{dk})} \right\},$$

$$I_{dki} \equiv \{ m : m \geq L_{dki} \} \text{ for } z_{di} = k \text{ and } \{ m : m \leq R_{dki} \} \text{ for } z_{di} \neq k. \quad (6)$$

We compute L_{dki} and R_{dki} in Eq. (6) as follows:

$$L_{dki} = \ln \frac{u_{dki} \sum_{k' \neq k} \exp(m_{dk'} + \lambda_{k's_d} + \tau_{k't_{di}})}{(1 - u_{dki}) \exp(\lambda_{ks_d} + \tau_{kt_{di}})},$$

$$R_{dki} = \ln \frac{\exp(m_{dz_{di}} + \lambda_{z_{di}s_d} + \tau_{z_{di}t_{di}}) - u_{dki} \sum_{k' \neq k} \exp(m_{dk'} + \lambda_{k's_d} + \tau_{k't_{di}})}{u_{dki} \exp(\lambda_{ks_d} + \tau_{kt_{di}})}. \quad (7)$$

Third, we consider the sampling of the λ_{ks} . We obtain the full conditional distribution as follows:

$$p(\lambda_{ks} | \boldsymbol{\lambda}^{-ks}, \mathbf{x}, \mathbf{z}, \mathbf{m}, \boldsymbol{\tau}, \boldsymbol{\mu}, \mathbf{K}_S) \propto \sqrt{(\mathbf{K}_S^{-1})_{ss}} \exp \left\{ -\frac{(\lambda_{ks} - \zeta_{ks})^2}{2(\mathbf{K}_S^{-1})_{ss}^{-1}} \right\} \cdot \prod_{\{d:s_d=s\}} \prod_{i=1}^{N_d} \frac{\exp(m_{dz_{di}} + \lambda_{z_{di}s} + \tau_{z_{di}t_{di}})}{\sum_k \exp(m_{dk} + \lambda_{ks} + \tau_{kt_{di}})}, \quad (8)$$

where $\zeta_{ks} \equiv -(\mathbf{K}_S^{-1})_{ss}^{-1} \sum_{s' \neq s} (\mathbf{K}_S^{-1})_{ss'} \lambda_{ks'}$. λ_{ks} depends on all other $\lambda_{k's'}$ for $k' \neq k$ and $s' \neq s$. The general slice sampler for λ_{ks} is given below.

- For $d \in \{d : s_d = s\}$, draw auxiliary variables u_{ks} and $u_{ksd1}, \dots, u_{ksdn_d}$:
 - Draw u_{ks} from $\mathcal{U} \left(\left\{ u : 0 \leq u \leq \sqrt{(\mathbf{K}_S^{-1})_{ss}} \exp \left\{ -\frac{(\lambda_{ks} - \zeta_{ks})^2}{2(\mathbf{K}_S^{-1})_{ss}^{-1}} \right\} \right\} \right)$.
 - Draw u_{ksdi} from $\mathcal{U} \left(\left\{ u : 0 \leq u \leq \frac{\exp(m_{dz_{di}} + \lambda_{z_{di}s} + \tau_{z_{di}t_{di}})}{\sum_k \exp(m_{dk} + \lambda_{ks} + \tau_{kt_{di}})} \right\} \right)$ for $i = 1, \dots, N_d$.
- Draw λ_{ks} from $\mathcal{U}(I_{ks} \cap \bigcap_{\{d:s_d=s\}} \bigcap_i I_{ksdi})$, where

$$I_{ks} \equiv \left\{ \lambda : \zeta_{ks} - \sqrt{-2(\mathbf{K}_S^{-1})_{ss}^{-1} \ln(\sqrt{(\mathbf{K}_S^{-1})_{ss}^{-1}} u_{ks})} \leq \lambda \leq \zeta_{ks} + \sqrt{-2(\mathbf{K}_S^{-1})_{ss}^{-1} \ln(\sqrt{(\mathbf{K}_S^{-1})_{ss}^{-1}} u_{ks})} \right\}, \text{ and}$$

$$I_{ksdi} \equiv \{ \lambda : \lambda \geq L_{ksdi} \} \text{ for } z_{di} = k \text{ and } \{ \lambda : \lambda \leq R_{ksdi} \} \text{ for } z_{di} \neq k. \quad (9)$$

In Eq. (9), we compute L_{ksdi} and R_{ksdi} as follows:

$$L_{ksdi} \equiv \ln \frac{u_{ksdi} \sum_{k' \neq k} \exp(m_{dk'} + \lambda_{k's} + \tau_{k't_{di}})}{(1 - u_{ksdi}) \exp(m_{dk} + \tau_{kt_{di}})},$$

$$R_{ksdi} \equiv \ln \frac{\exp(m_{dz_{di}} + \lambda_{z_{di}s} + \tau_{z_{di}t_{di}}) - u_{ksdi} \sum_{k' \neq k} \exp(m_{dk'} + \lambda_{k's} + \tau_{k't_{di}})}{u_{ksdi} \exp(m_{dk} + \tau_{kt_{di}})}. \quad (10)$$

Due to the space limitation, we omit the details of the sampling of the τ_{kts} , which can be derived in a similar manner to that of the λ_{ks} .

Other parameters are sampled as follows. The hyperparameters of the Gaussian process priors, i.e., σ_{S0} , σ_{S1} , l_S , σ_{T0} , σ_{T1} , and l_T , are sampled by

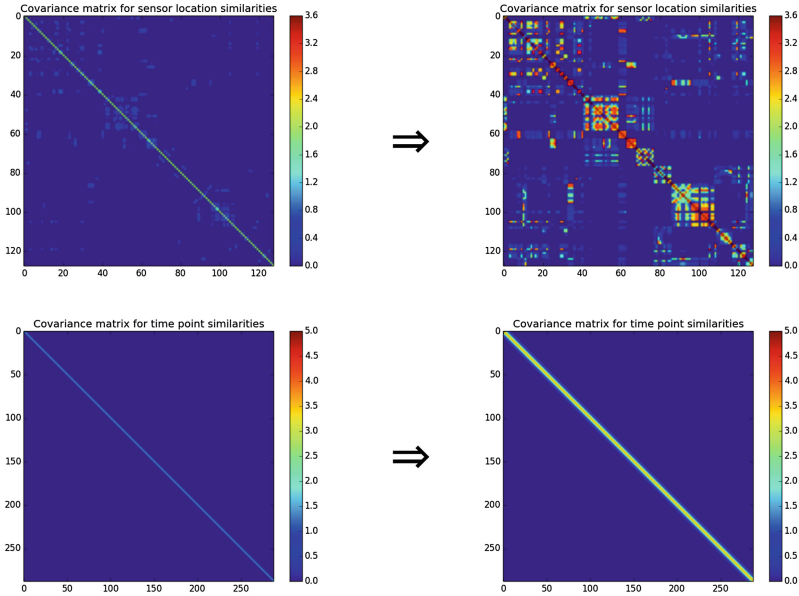


Fig. 2. Heatmap visualization of the covariance matrices representing the similarity between the sensor locations (on the top panel) and the similarity between the time points (on the bottom panel). On the left half of each panel, the initial entry values are given. After 800 iterations of MCMC for $K = 30$, we obtained an estimation of the entry values as given on the right half of each panel.

Metropolis Hastings algorithm. Figure 2 visualizes the result of the sampling. After hundreds of iterations, we obtained the covariance matrices where the similarity between sensor locations and that between time points were represented in a more emphasized manner in comparison to the initial matrices where all hyperparameters were set to 1. We sample the α_{ks} also by Metropolis Hastings. We set the shape and rate parameters of the gamma prior distribution $\text{Gamma}(a, b)$ to 1.5 and 1.0, respectively, because this gave a stably good result. While we tested Metropolis Hastings for a and b , it did not work. For μ and σ , we use the maximum-likelihood estimation to save the execution time.

4 Experiment

We evaluated the proposed method in terms of test data log likelihood by using the real data provided by New York City [1]. This data set consists of the traffic speeds measured by more than one hundred sensors placed along the streets. However, tens of sensors were reporting irregular data and were thus suspected of malfunction. Therefore, we chose 128 sensors. We normalized the data by regarding all measurements larger than 100 mph as 100 mph, though the number of such measurements was small. We regarded a set of traffic speeds measured by

the same sensor within 24 hours from 0 a.m. as a document. We built the data set from the traffic speeds measured within 21 days from May 27 to June 16 in 2013. For example, the sensor #289 produced the 21 documents, each corresponding to different days, as presented in Fig. 1. We thus obtained $21 \times 128 = 2688$ documents in total. The test data set was a randomly selected 20 percent of the traffic speed measurements in each document. A similar procedure for selecting test data is often adopted in an evaluation of topic models on text data. That is, a fixed percent of word tokens is selected randomly from each document and the likelihood is computed based on the inferred posterior distributions.

For our evaluation, the likelihood of the test data set was computed as follows:

$$p(\mathbf{x}_{test}) \equiv \sum_d \sum_{\{i:i \in testset\}} \sum_k \frac{\exp(m_{dk} + \lambda_{ks_d} + \tau_{kt_{di}})}{\sum_k \exp(m_{dk} + \lambda_{ks_d} + \tau_{kt_{di}})} \frac{\beta^{\alpha_k}}{\Gamma(\alpha_k)} x_{di}^{\alpha_k - 1} \beta^{x_{di} - 1}, \quad (11)$$

where $\beta \equiv a/b$. We set $K = 30$, because this setting gave a better or at least comparable likelihood when compared with other settings. We compared TRINH with the probabilistic model that is the same with TRINH except that the λ_{ks} s and the τ_{kt} s are set to 0. That is, we compared TRINH with the model using no location and timestamp data. The comparison was performed in terms of per-data log likelihood, i.e., $p(\mathbf{x}_{test})$ divided by the number of the traffic speed measurements contained in the test data set.

The comparison result is given in Fig. 3. The horizontal axis represents the number of iterations. The vertical axis gives the per-data log likelihood of the test data. A larger log likelihood is better, because this means that the inferred posterior distribution makes the test data set more probable. We show the results up to the 600th iteration. The log likelihood decreased almost monotonically for the remaining iterations. The line graphs present the mean of the per-data log likelihoods given by 20 inferences each starting from a different random initialization. The blue and orange graphs correspond to the result given by the model using no location and timestamp data and by TRINH, respectively. The error bar provides one standard deviation.

As Fig. 3 shows, TRINH could increase the log likelihood more rapidly than the probabilistic model using no metadata. To be precise, the mean log likelihood of our method was better than that of the compared method from the first to the 342th iteration. Further, the difference was larger than 1.0 until the 282th iteration. This may mean that the metadata enhanced the generalization capacity of the model. However, with respect to the best mean log likelihood achieved during the inference, TRINH was slightly worse. While TRINH gave -5.965 at the 368th iteration, the compared method gave -5.745 at the 409th iteration. At least for the data set used in this experiment, it can be concluded that TRINH fits the data quickly, but tends to overfit to the data.

This result can be explained as follows. It is likely that our method is quick in capturing a big picture of the distribution patterns of traffic speed data owing to the utilization of location- and time-dependencies. However, as inference proceeds, such dependencies may work as too strong a constraint to capture

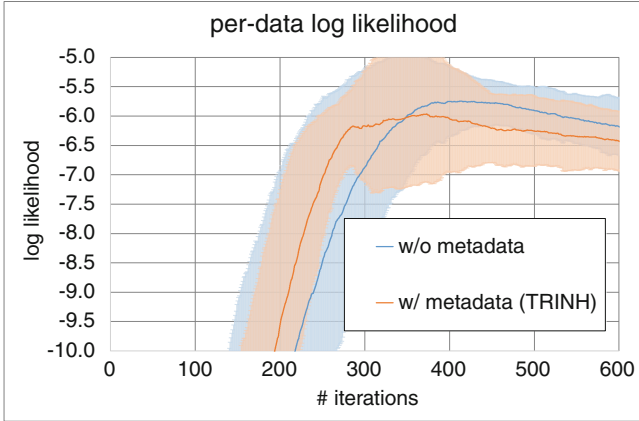


Fig. 3. The per-data log likelihood for each iteration. The horizontal axis gives the number of iterations up to 600. The vertical axis gives the per-data log likelihood, which is obtained by dividing the test data log likelihood by the number of test data. The test data set is a randomly selected 20 percent of the given data set.

important differences among traffic speed measurements. Therefore, it is an important remaining task to propose a method for controlling the influence of the location- and time-dependencies e.g. by introducing weighting coefficients for the λ_{ks} s and the τ_{kt} s.

5 Previous Work

There are important studies where the connection between LDA-like topic modeling and Gaussian process are considered. The model proposed by [16] utilizes a Gaussian process for obtaining the parameters of the gamma distribution, which in turn generates random numbers for determining per-document topic distributions. However, the authors use the Gaussian process for exploring the correlations between latent parameters. That is, the function drawn from the Gaussian process is evaluated at the locations in a latent space. In contrast, we utilize Gaussian processes to explore physical and temporal distances.

The topic model proposed by [11] draws functions from Gaussian process priors also for obtaining per-document topic probabilities. This work is similar to ours, because the logistic function is applied. However, we again do not use Gaussian process priors for modeling topic correlations, but for modeling spatio-temporal dependencies between topic probabilities. When compared to [16] and [11], it can be said that our work has a more practical motivation, because we need to encode our domain-specific knowledge in topic modeling.

The model devised by [2] uses a Gaussian process prior in an intricate manner. First, we draw as many functions as topics from the Gaussian process prior and evaluate them at the locations each corresponding to different documents.

Second, we construct a $K \times D$ matrix, whose rows are the per-document evaluations of the drawn function. Third, we use the columns of this matrix for obtaining per-document topic probabilities. However, such a complication leads to an inefficient inference, which prevented us to follow the proposal.

6 Conclusion

In this paper, we propose a method for traffic speed investigation with Bayesian hierarchical modeling, where we use Gaussian process priors for incorporating spatio-temporal nature of traffic speed measurements. Some proposals [4, 21] recommend to use more efficient methods for modeling random functions in place of Gaussian processes. The main reason is the computational complexity required for inference. Therefore, more efficient mechanisms for obtaining random functions may be adopted in our future work. Further, as we discussed in Sect. 4, it is also an important remaining task to propose a method for controlling the influence of the location- and time-dependencies.

References

1. Real-Time Traffic Speed Data, NYC OpenData. <https://data.cityofnewyork.us/Transportation/Real-Time-Traffic-Speed-Data/xsat-x5sa>
2. Agovic, A., Banerjee, A.: Gaussian process topic models. In: UAI, pp. 10–19 (2010)
3. Andzejewski, D., Zhu, X., Craven, M.: Incorporating domain knowledge into topic modeling via Dirichlet forest priors. *ICML* **382**(26), 25–32 (2009)
4. Bigelow, J.L., Dunson, D.B.: Bayesian semiparametric joint models for functional predictors. *J. Am. Stat. Assoc.* **104**(485), 26–36 (2009)
5. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. *JMLR* **3**, 993–1022 (2003)
6. Blei, D.M., Lafferty, J.D.: Correlated topic models. *NIPS* **18**, 147–154 (2005)
7. Chen, Z., Mukherjee, A., Liu, B., Hsu, M., Castellanos, M., Ghosh, R.: Leveraging multi-domain prior knowledge in topic models. In: *IJCAI*, pp. 2071–2077 (2013)
8. Chen, Z., Liu, B.: Topic modeling using topics from many domains, lifelong learning and big data. In: *ICML*, pp. 703–711 (2014)
9. Eisenstein, J., Ahmed A., Xing, E.P.: Sparse additive generative models of text. In: *ICML*, pp. 1041–1048 (2011)
10. Griffiths, T.L., Steyvers, M.: Finding scientific topics. *PNAS* **101**(Suppl 1), 5228–5235 (2004)
11. Hennig, P., Stern, D.H., Herbrich, R., Graepel, T.: Kernel topic models. In: *AIS-TATS*, pp. 511–519 (2012)
12. Masada, T., Takasu, A.: A topic model for traffic speed data analysis. In: Ali, M., Pan, J.-S., Chen, S.-M., Horng, M.-F. (eds.) *IEA/AIE 2014, Part II. LNCS*, vol. 8482, pp. 68–77. Springer, Heidelberg (2014)
13. Mimno, D., Wallach, H.M., Talley, E., Leenders, M., McCallum, A.: Optimizing semantic coherence in topic models. In: *EMNLP*, pp. 262–272 (2011)
14. Mimno, D., Wallach, H.M., McCallum, A.: Gibbs sampling for logistic normal topic models with graph-based priors. In: *NIPS Workshop on Graph Mining* (2008)

15. O'Connor, B., Stewart, B.M., Smith, N.A.: Learning to extract international relations from political context. In: ACL, pp. 1094–1104 (2013)
16. Paisley, J., Wang, C., Blei, D.: The discrete infinite logistic normal distribution for mixed-membership modeling. In: AISTATS, pp. 74–82 (2011)
17. Pan, B., Demiryurek, U., Shahabi, C.: Utilizing real-world transportation data for accurate traffic prediction. In: ICDM, pp. 595–604 (2012)
18. Rasmussen, C.E., Williams, C.K.I.: Gaussian Processes for Machine Learning. MIT Press, Cambridge (2006)
19. Robert, C.P., Casella, G.: Monte Carlo Statistical Methods. Springer, New York (2004)
20. Salakhutdinov, R., Hinton, G.E.: Replicated softmax: an undirected topic model. NIPS **22**, 1607–1614 (2009)
21. Scarpa, B., Dunson, D.B.: Enriched stick-breaking processes for functional data. J. Am. Stat. Assoc. **109**(506), 647–660 (2014)
22. Srivastava, N., Salakhutdinov, R., Hinton, G.E.: Modeling documents with deep boltzmann machine. In: UAI (2013)
23. Xu, Z., Chen, M., Weinberger, K.Q., Sha, F.: From sBoW to dCoT: marginalized encoders for text representation. In: CIKM, pp. 1879–1884 (2012)

An Effective Approach to Background Traffic Detection

Quang Tran Minh^(✉)

Hochiminh City University of Technology,
268 Ly Thuong Kiet, Hochiminh City, Vietnam
quangtran@hcmut.edu.vn

Abstract. Background (BG) traffic detection is an important task in network traffic analysis and management which helps to improve the QoS and QoE network services. Quickly detecting BG traffic from a huge amount of live traffic travelling in the network is a challenging research topic. This paper proposes a novel approach, namely the periodicity detection map (PDM), to quickly identify BG traffic based on periodicity analysis as BG traffic is commonly periodically generated by applications. However, it is not necessary that every BG traffic flow is periodic, hence the periodicity analysis based approaches cannot detect non-periodic BG flows. This paper also discusses the capability of applying a machine learning based classification method whose training dataset is collected from the results of the PDM method to solve this issue. Evaluation analysis and experimental results reveal the effectiveness and efficiency of the proposed approaches compared to the conventional methods in terms of computational costs, memory usage, and ratio of BG flows detected.

Keywords: Background/foreground traffic · Traffic analysis · Periodicity · Periodicity detection map · Machine learning

1 Introduction

The revolution of smart mobile devices specially smart phones, sensors and wireless technologies such as WiFi, 3G, WiMAX, LTE networks, etc., allow sophisticated applications and services like social networks, M2M communications, IoT applications, smart cities, etc., to be realized. Consequently, a huge amount of data is generated and carried in computer networks consuming the scarce radio and bandwidth resources, resulting in network congestion or failure. In fact, it is not necessary that every traffic is directly related to user activities. Concretely, a large amount of data is generated automatically by applications on mobile devices to update their status. This background (BG) traffic is contrasted with the foreground (FG) traffic which is generated by users in particular communication operations.

In practice, if the BG and FG traffic are separated effectively, network operators can place a suitable policy to control the network traffic to improve the

QoS and QoE of network services. For instance, the delivery of BG traffic at the peak time can be delayed to save the network resources for FG traffic to serve user's communications need. Existing work has been proposed to detect BG traffic by analyzing user activities based on the device screen status detection. These researches assume that, if the data is generated/received at the devices network interface while its screen is off, the traffic would be BG traffic. This approach is useful for battery saving, improving user experiment services, and so on, which are deployed on individual devices [1, 2]. However, it is not suitable for traffic control to improve the network QoS. In order to place any network optimization, traffic condition must be recognized. Therefore, the BG/FG detection mechanism should be deployed at the network edge such as on ISP routers. Nevertheless, this approach faces on an emerging issue of processing a huge amount of traffic traveling on the network without any information about user activities. This makes the BG/FG separation harder to be resolved.

Moreover, as the privacy legislation in telecommunication must be strictly followed, deep packet inspection (DPI) related methods are not applicable. To overcome this difficulty, several researches have proposed to apply machine learning (ML) methods to analyze the statistical data of traffic flows. However, ML-based approaches are time consuming and dependent on relevant training datasets which are not always available in advance.

Under preliminary researches and observations on real traffic flows [3], we revealed that BG traffic is commonly periodically generated as applications periodically communicate the servers to update their status. This trait can be leveraged to quickly detect the explicit BG traffic. However, it is not necessary that every BG traffic flow is periodical as applications may synchronize their data with the server based on events. For example, the Drop Box application on a device will update figures to the Drop Box server at the time they are taken. To detect non-periodic BG traffic, an ML-based method would be helpful. Fortunately, the BG traffic flows detected previously by a periodicity analysis method can be used as training examples for the ML model.

This paper proposes an effective approach to separate BG traffic with two phases: (1) phase 1 is to quickly detect the clear BG traffic flows based on periodicity analysis, and (2) phase 2 is to utilize an ML model trained by examples detected in phase 1 to classify non-periodic flows. The experimental evaluations and analysis reveal that this approach is not only robust for BG/FG traffic separation but it is also possible to be deployed in the real system.

The rest of the paper is organized as follows: Sect. 2 reviews the related work. The overall architecture and problem formulation are presented in Sect. 3. Section 4 describes the proposed approaches, while Sect. 5 presents evaluation results and analysis. Section 6 concludes this paper.

2 Related Work

BG traffic is traffic generated by applications on devices to maintain their network connectivity such as network management packets (ARP, DHCP,

IMCP,...), network service handshakes (NetBIOS, DNS,...), or applications heartbeats (Windows update, Yahoo weather update, ...) [1–3]. In contrast, FG traffic is generated by users on real communications such as web-surfing, making a phone call, etc. As BG traffic is not immediately useful for users, it can be delayed if needed to save the device (battery, memory, computational capability,...) or network (bandwidth, radio channel,...) resources at the critical times.

As BG/FG traffic separation can help to optimize the network administration and management, especially to improve the network quality, several researches have been proposed to find a sound solution to quickly identify BG traffic from a vast traffic flows on computer networks [4,5]. The study in [4] provides an evaluation on the effect of BG traffic on application and protocol behaviors by quantifying the interaction of applications such as HTTP, multimedia applications with a variety of BG traffic models. Kenesi et al., analyze the impact of BG traffic on TCP throughput [5]. These analysis support the motivation of our work on BG traffic detection towards a sound BG traffic management and control solution to improve the quality of network services.

User activity analysis is a potential direction to detect BG traffic. User activity involving inferred from the screen state (on/off) can be associated with traffic generated/received by the device’s network interface to detect BG traffic [1,2]. NetSense [6] and LiveLabs [7] are two interesting live projects that study user activities on smart phones including social networking, location-based services, screen state, etc. The essential difficulty of this method is that a specific application must be installed in user equipment (UE). This not only creates stressfulness to users but also requires more resources and computational cost. Moreover, BG traffic detected on UEs would be useful for optimizations on individual devices (e.g., battery saving), while it would not be enough for optimizing the whole network. In order to optimize network services applying traffic engineering techniques, information about the whole network load and the amount of BG traffic flows are needed. This work resolves this issue by proposing a novel method deployed at the network edge to quickly capture, analyze and optimize a large amount of network traffic, without imposing any policy on the UE.

Studies in [1,2] and our previous work [3], revealed that BG traffic generates a lot of periodic network maintenance traffic. This leads to a huge amount of BG traffic traveling in the networks as each device may concurrently run many background applications. It is necessary to quickly identify these BG traffic, thereby they can be delayed to yield the scarce resources, especially at the peak time, to the user-oriented traffic (FG traffic). This study leverages the periodical characteristic of BG traffic to quickly detecting the explicit BG traffic flows. The difficulty here is that traffic flows are commonly long and sparse time-series data. They require a huge computational time for analyzing [8,9]. This work proposes a novel method, namely the Periodicity Detection Map (PDM), that efficiently works with long and sparse traffic flows, hence significantly reduces the computational cost to $O(n)$, when the analyzed data is long and sparse. In addition, as not every BG traffic flow is periodic, the PDM may not be able to detect non-periodic BG flows. This paper also proposes to apply an

ML-based classification model that utilizes the results of the PDM method as training examples to solve this issue. As a result, the proposed approaches not only quickly detect the explicit BG traffic but can also effectively identify non-periodic BG traffic flows.

3 Overall Architecture and Problem Definition

The overall architecture of the proposed BG/FG traffic separation system is depicted in Fig. 1. The system is deployed at the network edge which is in between the wireless radio access network (RAN) and the core network as shown in Fig. 1a. The periodicity of the captured traffic is immediately examined (Fig. 1b) to quickly identify whether it is BG traffic or not using the periodicity analysis. For non-periodic traffic flows, their statistical data such as average number of packets per TCP connection (upward and downward) of the traffic flows, is analyzed by a ML-based classification model to classify its traffic type (BG or FG). The classified data is stored in a database which is used as an adaptive training dataset for the classification model.

In order to clarify the BG/FG traffic separation model, we start with following definitions.

Definition 1: *BG traffic is traffic generated by applications installed on the UE without any activity from the user. FG traffic is traffic generated by user activities.*

According to studies in [3], BG traffic is periodically generated by applications. To keep track of data generation for periodicity analysis we define flows of data communications based on the occurrences of TCP connections as follows:

Definition 2: *Flow is a series of time stamps (in second) representing the occurrence of TCP connections that involve to a particular mobile device and particular application/service on a server.*

Based on this definition, flows are separated by a tuple of $\{\text{source IP, destination IP, destination port}\}$, denoted as $\{\text{srcIP, destIP, destPort}\}$. Different to existing approaches, source port is not used for flow identification since applications may use non-registered source ports to avoid port confliction or traffic control from network operators. As defined, each flow is a time series denoted as $T = e_0, e_1, e_2, \dots, e_{n-1}$, where $e_i = \{1|0\}$ is a data element representing the occurrence (“1”) or non-occurrence (“0”) of a TCP session at time t_i .

As mentioned, BG traffic is periodically generated, the proposed approach should immediately analyze this property of the flows in its 1st phase to quickly identify whether the involving traffic is BG traffic or not. The difficulty here is that traffic flows are commonly larges (while sparse) leading to a large computational time is required for analyzing the flow’s periodicity, as the conventional methods raises a complexity of $O(n^2 \log n)$, where $n = |T|$ [9].

In addition, it is not necessary that every BG traffic is periodically generated since applications may arbitrarily synchronize with servers in accordance with some events. For example, the Drop Box application on a device may initiate data

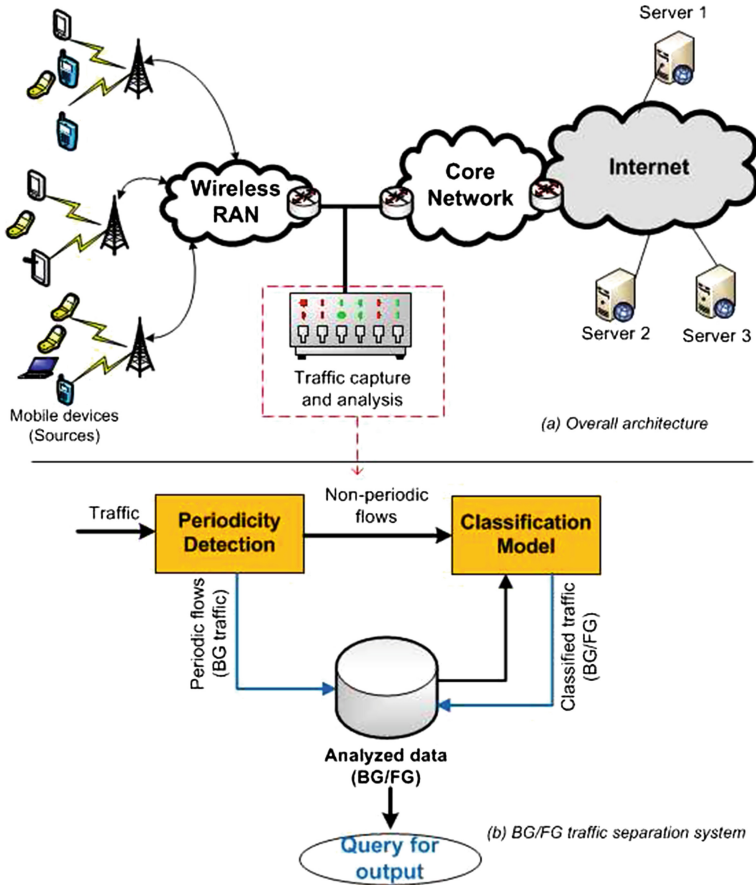


Fig. 1. The overall architecture and design of the proposed BG/FG traffic separation system

synchronization when photos are taken. Consequently, BG traffic detection using periodicity analysis may not properly work with this situation. To overcome this issue, the system applies an ML-based classification model in its 2^{nd} phase to deal with non-periodic traffic flows. Fortunately, this model can be trained by the dataset resulted from the 1^{st} phase. Details of the two-phase BG/FG traffic separation approach are presented in the next section.

4 The Proposed Approach

4.1 Periodicity of a TCP Connection Flow

In the 1^{st} phase, the proposed BG/FG traffic separation system will detect the periodicity of the occurrences of TCP connections in considered flows. Concretely, an effective mechanism for symbolical periodicity detection is proposed,

where the symbol to be detected is “1” representing the occurrences of TCP sessions [9].

In order to validate a periodicity, three fundamental information should be provided: the period p , the starting point s and the confidence c . Obviously, the combination of (p, s, c) would be exponentially large when processing a long time series T . This reveals a huge computational cost for periodicity detection. The following sub-sections analyzes the existing mechanisms for periodicity detection and proposes a novel method that can quickly detect the periodicity of traffic flows to significantly reduce the computational cost.

4.2 Auto Correlation (AC) and Projection Based Approaches

One of the conventional mechanisms to examine the periodicity of a time series T is calculating its self-convolution using auto-correlation function (ACF) [10] as shown in (1)

$$r_T(l) = \frac{1}{n-l} \sum_{k=0}^{n-1-l} T(k)T(k+l) \quad (1)$$

where, $l = 0, 1, 2, \dots, n/2$ is the *lag* (the shifting value) and $n = |T|$. The maximum $r_T(l)$ is selected since it represents the highest potential for periodicity with $p=l$. This approach raises a complexity of $O(n^2)$ while it just examines periodicities in accordance with the starting point at $T/0$. Meanwhile, the periodicity may start at a latter starting point $i(0 < i < n-2)$ revealing that all sub-series $T_i = e_i, e_{i+1}, e_{i+2}, \dots, e_{n-1}$ must be applied to the ACF recursively, increasing the complexity to $O(n^3)$.

The Fast Fourier Transform (FFT) [10] can be applied to reduce the complexity of self-convolution to $O(n \log n)$ revealing $O(n^2 \log n)$ for an exhaustive process of all the potential cases with different starting points as mentioned above. This complexity is still huge in real world applications where T is commonly long. As a result, an effective method to analyze a large amount of long traffic flows is crucial.

In addition to the AC-based method, several works propose a more intuitive method based on the projection as presented in [9] as follows:

Definition 3: *The projection of a time series T according to period p starting at position s , denoted as $\Pi_{p,s}(T)$, is:*

$$\Pi_{p,s}(T) = e_s, e_{s+p}, e_{s+2p}, \dots, e_{s+(m-1)p} \quad (2)$$

where, $0 \leq s < p, m = \lfloor (n-l)/p \rfloor, n = |T|$.

Definition 4: *The confidence c of a periodic symbol e (e.g., $e=“1”$) in T according to period p starting at s represents how relevant this periodicity happens in T , and is defined as:*

$$c = \frac{\sum z_i}{|\Pi_{p,s}(T)| - 1} \quad (3)$$

where, z_i is the number of distances between consecutive “1” at any position i in $\Pi_{p,s}(T)$.

For example, given $T=011001001101101$, we obtain $\Pi_{3,0}(T)=00011$ resulting in $c=1/4=25\%$, while $\Pi_{3,2}(T)=11111$ resulting in $c=4/4=100\%$. In this example we can conclude that T is periodic with $p=3$, $s=2$, and $c=100\%$ (i.e., “1” occurs at every interval $p=3$ starting at $T/2$ or e_2).

The advantage of this approach is that it is quite simple and intuitive for implementation. However, it requires $O(n^3)$ computation time, in the worst case, to detect the periodicity of a time series T of length n . In the next section, we propose a novel approach that quickly detects the periodicity of long traffic flows.

4.3 Periodicity Detection Map (PDM)

As the periodicity of a time series T should be quickly detected to identify whether the corresponding traffic flows is an explicit BG flows, we can simplify our solution to quickly identify the periodicity with the potentially highest confidence c that satisfies a threshold τ (e.g., $\tau = 80\%$). This section presents a novel mechanism, namely the PDM, which is specifically effective with long and sparse time series such as TCP connection flows discussed in this work. For example, with an 24-h captured data, T 's length is 86,400 (points) if the granularity of discretization is 1 s while it may consist of a few TCP sessions (element “1”). The PDM leverages the sparseness of traffic flows to store and process only involving data, hence significantly reduces the computational complexity as described below.

An PDM is a map that converts a time series T into a specific data structure which is useful for quickly identify the periodicity in T . PDM is created for each time series T as a table of (per, pos) . Here pos is the list of positions in T where TCP connections occur and per (where $1 < per < n/2$) is the potential period in T . Pos and per are column and row labels of the PDM, respectively. For each cell identified by a pair of (per, pos) , two values are calculated: (i) the identifier that identifies a group of potential positions (pos) contributing to the periodicity whose period is per , we denote this value per_group ; and (ii) the position of the pos in the $\Pi_{per,s}(T)$, where s is the first value of pos in the same per_group , we denote this value per_pos . These values are calculated in (4) and (5).

$$per_group(per, pos) = pos \pmod{per} \tag{4}$$

$$per_pos(per, pos) = pos \div per \tag{5}$$

For example, the PDM of $T=011001001101101$ ($n=15$) is represented in Table 1 and described as follows.

Firstly, $pos=\{1,2,5,8,9,11,12,14\}$ and $per=[1:7]$ representing column and row labels of the PDM. For each cell denoted by (per, pos) a pair of (per_group, per_pos) is calculated. For example, at cell $(per=2, pos=5)$ the (per_group, per_pos) is (1, 2), represented as $\frac{1}{2}$ in the table. It should be noted

Table 1. The PDM of time series $T=011001001101101$

$\frac{pos}{per}$	1	2	5	8	9	11	12	14	per_size	c
1	$\frac{0}{1}$	$\frac{0}{2}$	$\frac{0}{5}$	$\frac{0}{8}$	$\frac{0}{9}$	$\frac{0}{11}$	$\frac{0}{12}$	$\frac{0}{14}$	14	3/14
2	$\frac{1}{0}$	$\frac{0}{2}$	$\frac{1}{2}$	$\frac{0}{4}$	$\frac{1}{4}$	$\frac{1}{5}$	$\frac{0}{4}$	$\frac{0}{4}$	7	1/7
3	$\frac{1}{1}$	$\frac{2}{0}$	$\frac{2}{1}$	$\frac{2}{2}$	$\frac{0}{2}$	$\frac{2}{3}$	$\frac{0}{4}$	$\frac{2}{4}$	4	4/4
4										
5										
6										
7										

that for each value of per (each row) there are several per_groups . For example, there are three per_groups , namely $\{0, 1, 2\}$, in accordance with $per=3$.

As mentioned before, a per_group represents a group of potential positions that contribute to the periodicity whose period is per , the per_group with the largest length (in accordance with the given per) is the most important (compared to other per_group in the same row) for examining the periodicity of T with a period per . Let denote $per_group_List(per)$ the largest per_group in accordance with per , and $per_pos_List(per)$ the list of corresponding per_pos . It should be noted that, for the legibility only the per_pos which is in the $per_pos_List(per)$ is presented in the PDM. For example, the $per_group_List(3)=\{2, 2, 2, 2, 2\}$ and $per_pos_List(3)=\{0, 1, 2, 3, 4\}$.

With this design, the PDM can be effectively used to quickly detect the most potential period of the given time series T with following advantages:

- Periodicities with short periods are examined first. This is practically useful since the periodicity with a long period can be inferred from such a short periodicity, but the reversed inference is not necessary to be correct. For example, if T is periodic with $per=3$, and confidence $c=100\%$ then it is straight forward to conclude that T is also periodic with $per=6$ and $c=100\%$. Obviously, it should be careful for an inference in the reversed case.
- The algorithm is stopped when the confidence c satisfies the threshold τ
- Confidence c is calculated directly from the $per_group_List(per)$ and the $per_pos_List(per)$, requiring no additional computation time for extracting the real periods in T .

The confidence of a periodicity with $period=per$ in T is calculated as follows:

Let denote $per_size(per)$ the number of (expected) intervals obtained by dividing T by per as in (6).

$$per_size(per) = \left\lfloor \frac{n}{per} \right\rfloor \tag{6}$$

Let denote $z(per)$ the accumulative counts of the continuous occurrences in T in accordance with period per . This value can be counted directly from

$per_pos_List(per)$. For example, with $per_pos_List(2)=\{0,2,4,5\}$, $z(2)=1$ (applied to 4 and 5 in the list); with $per_pos_List(3)=\{0,1,2,3,4\}$, $z(3)=4$.

The confidence in accordance with period per , denoted as $c(per)$ is calculated in equation (7).

$$c(per) = \frac{z(per)}{per_size(per)} \quad (7)$$

Table 1 shows the number of (expected) intervals per_size and confidence c in accordance with a specific period per in its last two columns. As presented, the algorithm can provide a solution and stops after three iterations ($per=3$) when $c(3)$ reached 100%. In another word, the algorithm can quickly conclude that the given time series $T=011001001101101$ is periodic with period $per=3$ and confidence $c=100\%$. The effectiveness of this method will be evaluated and discussed in the next section.

4.4 BG Traffic Detection Using Machine Learning Approach

As mentioned in Sect. 1, it is not necessary that every BG traffic flow is periodical as applications may synchronize their data with the servers based on events. Consequently, the proposed PDM may not be able to detect non-periodic BG traffic flows. As discussed in Sect. 3 and Fig. 1, to solve this issue we apply an ML-based classification approach to learn the statistical features of the BG traffic to examine non-periodic flows. Concretely, the statistical data of BG traffic detected by the PDM method (via periodicity analysis) is used to train the ML model which is then used to classify non-periodic traffic flows.

Obviously, any ML model such as ANN, Naive Bayer, or decision tree (J48) [11], and so on, can be used as a classifier. In order to build the classifier, 28 statistical features extracted from traffic flows are utilized as follows: *Minimum*, *Maximum*, *Average*, and *Standard deviation* of the *packet size*, the *amount of data per TCP session*, the *number of packet per TCP session*, and the *duration* (in seconds) of TCP sessions. It should be noted that all the features, excepted the *duration*, are calculated in both *upward* and *downward* directions. Another notice is that the *periodicity* of flows is not used for training to avoid any bias from over fitting. The evaluation of this method is presented in the next section.

5 Evaluation

This Section evaluates and analyzes the effectiveness and the efficiency of the proposed approaches in the comparison with conventional methods in terms of computational cost, memory usage, and ratio of BG flows detected.

5.1 Evaluation Environment

Both experimental and commercial data were collected for evaluation. The experimental data were obtained with a limited number of mobile devices and particular environment settings. In this setting BG and FG traffic data were collected

Table 2. Installed Applications for BG traffic generation

Application Name	Category
Android 4.0.3	OS
Yahoo! Topics	News & Magazines
Nikkei	News & Magazines
gReader (Google Reader)	News & Magazines
AccuWeather Platinum	Weather
Rain alarm	Weather
BeWeather & Widgets	Weather
Dropbox	Productivity
Evernote	Productivity
Google Drive	Productivity

Table 3. Characteristics of the analyzed datasets

Dataset	No. of TCP sessions	No. of flows	Periodic ratio
1hrBG	6,300	127	85 %
Browse	2,029	222	8 %
Real	213,422	6,101	23 %

separately, hence serving as ground truth data. Concretely, two scenarios were setup: (a) devices were left without any user action thereby only BG traffic was generated during an hour by applications or OS installed in mobile devices (shown in Table 2); we named this *1hrBG* dataset; and (b) users browsed the Web for an hour while all of the aforementioned BG applications were uninstalled to mostly generate FG traffic (we named this *Browse* dataset). The commercial data was collected from the real cellular network during 1 h in which the first 50 users involving the largest number of TCP connections were selected. We named this *Real* dataset.

5.2 BG Traffic Flows Detected by the Proposed Methods

This section evaluates the capability of the proposed methods on BG traffic detection. Table 3 shows the characteristics of the different datasets studied in this work, namely the number of TCP sessions, and the number of flows on *1hrBG*, *Browse*, and *Real* datasets. The last column shows the ratio of BG traffic flows detected by the PDM method (via the periodicity detection). As shown, this ratio is high in the *1hrBG* dataset while it is low in the *Browse* and the *Real* counterparts. This fact confirms our hypothesis that the periodic flows commonly come from BG traffic.

However, as discussed in Sect. 4.4, the PDM may miss non-periodic BG traffic flows. This drawback can be overcome by an ML-based classification model,

namely the decision tree (J48) [11], in this work. We extracted 28 statistical features discussed in Sect. 4.4 from the periodic flow in the *1hrBG* dataset (known as BG flows) and non-periodic flows on the *Browse* dataset (known as FG flows) to train the classification model. After that, the model is used to detect the BG traffic from non-periodic flows in the *Real* dataset. The experimental results reveal that further BG traffic flows are detected from non-periodic flows resulting in a total of 31 % (23 % by the PDM and further 8 % by the J48) BG traffic flows were detected in the *Real* dataset. Interestingly, this ratio is compatible with a claim in [2] that 1/3 of traffic on the 3G network (recorded at the mobile phones network interfaces) is BG traffic.

5.3 Complexity Analysis for the PDM Method

This sub-section analyzes the complexity of the PDM method in the comparison with conventional methods. As discussed in Sect. 4.3, the PDM method consists of 3 steps whereby the complexity is analyzed as follows:

Step 1: Obtain the list of *pos* (of size m , where $m \ll n$) from T . This step raises a complexity of $O(n)$.

Step 2: For each *per* (each row in the PDM table), compute *pos_group*, *per_pos* at each (*per*, *pos*) cell, and calculate the confidence $c(per)$. As the length of each PDM row is m and the computation time for each task mentioned above is a constant, the computation time required for each *per* is $O(m)$.

Step 3: Stop if $c(per) > \tau$. Otherwise, repeat step 2.

Let denote i ($1 \leq i \leq n/2$) the number of iterations before the algorithm successfully stops and provides a solution, the complexity would be:

$$Complexity = O(n) + O(i * m) \quad (8)$$

When the analyzed data is long and sparse $m \ll n$, equation (8) can be simplified as:

$$Complexity = \begin{cases} O(n), i \ll n \\ O(n) + O(m * n/2) = O(n), i = n/2 \end{cases} \quad (9)$$

As shown, when the analyzed data is long and sparse ($m \ll n$) as the dataset for TPC connection flows in this work, the complexity of the PDM method is $O(n)$. This complexity is significantly smaller than that of the conventional approaches, namely $O(n^3)$ in the AC, and $O(n^2 \log n)$ in the FFT or the projection-based approaches [9].

6 Conclusion

This paper proposed a novel method to quickly identifying the BG traffic flows based on the analysis of their periodicity. The proposed PDM method effectively identify the list of occurrence (namely the *pos*) of TCP sessions in the given traffic

flow T . This list is commonly significantly smaller than the original flow, hence reduce the computational space. The PDM also provides a capacity to agilely provide an answer of whether the flow is periodic or not. Consequently, the PDM significantly reduces the complexity from $O(n^2 \log n)$ in the conventional approaches to $O(n)$, when the analyzed data is long and sparse.

This paper also discussed the capability of using the output resulted by the PDM as training dataset for further analysis using ML-based method. Concretely, the ML-based classification model such as the J48 can be trained by the PDM output to further identify the non-periodic traffic flows. Evaluations on both the experimental and commercial data confirm the effectiveness and efficiency of the proposed approaches (both the PDM and the ML-based methods).

In practice, however, traffic flows may contain both the BG and the FG traffic. Further analysis to identify these traffic flows is a challenging task. In the future, more detailed performance will be evaluated in order to confirm the robustness and the flexibility of the proposed approaches. Putting these approaches into realization with optimization strategies for a robust network traffic management system is also an interesting research direction.

References

1. Huang J., Qian F., Mao Z. M., Sen S., Spatscheck O.: Screen-off traffic characterization and optimization in 3G/4G networks. In: Proceedings of the IMC 2012, Boston, Massachusetts, USA, pp. 357–364 (2012)
2. Meng L., Liu S., Striegel A.D.: Characterizing the utility of smartphone background traffic. In: Proceedings of the 23rd International Conference on Computer Communication and Networks (ICCCN), Shanghai, China, pp. 1–5 (2014)
3. Quang T. M., Koto H., Ano S., Chen L., Arakawa S., Murata M.: Proposal of Periodicity Detection Method for Separation of Background Traffic. IEICE Technical report, MoNA2014-55, vol. 114, no. 308, pp. 37–42 (2014)
4. Vishwanath K.V., Vahdat A.: Evaluating distributed systems: does background traffic matter? In: USENIX Annual Technical Conference, pp. 227–240 (2008)
5. Kenesi Z., Szabo Z., Belicza Z., Molnar S.: On the effect of the background traffic on TCPs throughput. In: Proceedings of the 10th IEEE Symposium on Computers and Communications, pp. 631–636 (2005)
6. NetSense. <http://netsense.nd.edu/>. Accessed Jun 2015
7. LiveLabs. <http://centres.smu.edu.sg/livelabs/>. Accessed Jun 2015
8. Berberidis, C., Vlahavas, I.P., Aref, W.G., Atallah, M.J., Elmagarid, A.K.: On the discovery of weak periodicities in large time series. In: Elomaa, T., Mannila, H., Toivonen, H. (eds.) PKDD 2002. LNCS (LNAI), vol. 2431, pp. 51–61. Springer, Heidelberg (2002)
9. Elfeky, M.G., Aref, W.G., Elmagarid, A.K.: Periodicity detection in time series databases. IEEE Trans. Knowl. Data Eng. **17**(7), 875–887 (2005)
10. Knuth, D.: The Art of Computer Programming. Computer Science and Information Processing, vol. 2, Second edn. Addison-Wesley, Reading (1981)
11. Nguyen, T., Armitage, G.: A survey of techniques for Internet traffic classification using machine learning. IEEE Commun. Surv. Tutorials **10**(4), 56–76 (2008)

An Approach for Developing Intelligent Systems in Smart Home Environment

Tran Nguyen Minh-Thai^(✉) and Nguyen Thai-Nghe

College of Information and Communication Technology, Can Tho University,
3/2 Street, Can Tho City, Vietnam
{tnmthai, ntnghe}@cit.ctu.edu.vn

Abstract. Smart home systems are taken into account recently. By detecting abnormal usages in these systems may help users/organizations to better understand the usage of their home appliances and to distinguish unnecessary usages as well as abnormal problems which can cause waste, damages, or even fire. In this work, we first present an overview on the Smart Home Environments (SHEs) including their classification, architecture, and techniques which can be used in SHEs, as well as their applications in practice. We then propose a framework including methods for abnormal usage detection using home appliance usage logs. The proposed methods are validated by using a real dataset. Experimental results show that these methods perform nicely and would be promising for practice.

1 Introduction

Recently, with the concern of electricity conservation, one of the important applications is abnormal usage detection of appliances. Due to the significant efforts in reducing the emissions of CO₂ and other GHGs (greenhouse gases), many researchers have focused on the electricity conservation in the residential sector. Abnormal usage detection can help residents not only reducing electricity consumption, but also having benefit for the environment. However, previous researches [1, 2] have been focused on analysis of the usage behavior on single device and neglect of the appliance correlations. Actually, the correlation among the usage of some appliances can provide valuable information to assist residents better detect abnormal usage of their appliances.

Abnormal usage of the energy consumption for a particular period is significantly different than that of the previous time, during which some appliances are unexpectedly operating. Appliances in a frequent sequence also show the correlation among the devices based on their locations in a house. The correlation among the usage of some appliances can provide valuable information to assist residents better understand how they use their devices.

In fact, detection of abnormal usage is an important issue in smart home research. However, this is a challenging task when designing a remarkably effective and computationally reasonable solution. Our appliance behavior usage usually varies according to different periods of time and season, i.e. many behaviors of the same appliances in summer and in winter are totally different. For instance, a heater can be used daily in winter, but is seldom turned on in summer. In contrast, an air conditioner is usually

turned on in summer, but is almost never used in winter. Appliances also have unique patterns such as seasonal types or daily types; for example, while the heater, a seasonal appliance, is frequently operated only in the summer; the light, a daily appliance, is usually turned on and off every day. Figure 1 illustrates an example of abnormal usage detection problem.

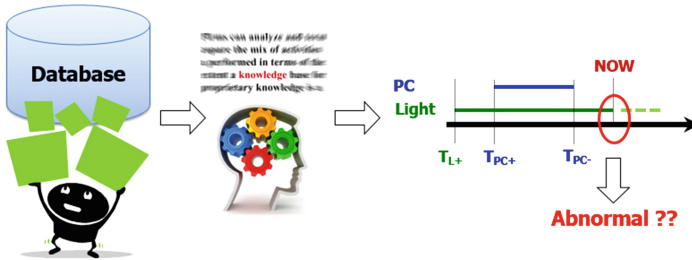


Fig. 1. An example of abnormal usage detection.

The consideration of correlation among appliance is a challenge issue for anomaly detection. Since the usage of a device has duration time, the correlations among appliances can be treated as an interval sequence. The abnormal usage based on the interval sequence is significantly different than that of the previous researches which only includes the information of a single appliance.

In this work, we first review some definitions and related problems in the Smart Home Environments (SHEs) such as their classification, their structure/architecture, and techniques which can be used in SHEs, as well as their applications to provide the reader a complete understanding in the areas of SHEs. We then present a framework with several methods which can be used to detect abnormal usage of home appliances in smart homes. These methods take into account the appliance correlations. The first method is used for Calculating Anomaly Score using Extreme Value Theory. The second method is used for Calculating Anomaly Score based on Sequence Patterns, and the last one is used for Determining Abnormal Time Intervals.

2 Smart Home Environments - SHEs

The terms smart homes, intelligent homes, home networking have been used for more than a decade to introduce the concept of networking devices and equipment in the house. There are several definition of smart home environments in research areas. The best definition of smart home environments is the integration of technology and services through home networking for a better quality of living [3].

2.1 SHEs Classification

SHEs are complex environments with its three separable interest areas as follows [3]:

- (i) A Home Automation System with a set of home appliances such as washing and cooking machines, refrigerators, heaters, thermometers, lighting system, power outlets, energy meters, smoke detectors, televisions, game consoles and other entertainment devices, windows and doors controllers, air conditioners, video cameras.
- (ii) A Control System that combines human with software use the information collected from the sensors.
- (iii) A Home Automation Network makes all the smart home environments components, including the Home Automation System and the Control System, can change status and control information.

The most recent review of SHE is delivered in [4]. Its authors focused mainly on SHE projects, as well as on the SHE building blocks including components, devices and networks. However, their paper lacks a classification of SHEs from the point of view of their application areas. SHEs are consisted of a number of hardware and software components. SHE can be classified according to many different criteria such as structure, architecture, middleware, application, and computational methods.

2.2 Structure and Architecture

A SHE is consisted of three systems: Home Automation System, Control System and Home Automation Network. Each system has its hardware and software components.

The Home Automation System may contain a large variety of home appliances which depend on specific applications. A new concept, smart object describes advanced devices in smart home, which is made up of three important parts: the physical part, the hardware infrastructure, and the software layer.

The Home Automation Network is consisted of physical technology and communication protocols. Powerline, busline, and wireless are three main classes of home network technologies.

The most complex part of a SHE is the Control System. It provides reactive behavior to specific events such as smoke detection, temperature variation. A lot of methods were proposed for the design and development of sophisticated Control Systems of SHE using results of artificial intelligence, multi-agent systems and automation control.

The architecture of SHE is unequivocally impacted by the computational abilities of their segments. A large portion of the computational necessities of SHE are identified with the accomplishment of the elements of its Control System. They identify two main architectural styles of SHE: centralized, and distributed.

In a centralized SHE architecture, the Control System is acknowledged by method for a computer system that is mindful with information obtaining from sensors, client interfacing, and additionally with the usage of control calculations and sending guidelines to actuators.

In a distributed SHE architecture, the software of the Control System is conceptualized and actualized as a disseminated processing framework. The distributed architecture profits by the computational resources of smart objects to embed software components into the nodes of the Home Automation Network.

2.3 SHEs Techniques

SHE ended up being a productive experimentation ground for the assortment of computational methods and techniques proposed by artificial intelligence and multi-agent systems communities. They can be generally arranged along two orthogonal tomahawks into: (i) centralized and (ii) distributed approaches, as well as (i) symbolic and (ii) sub-symbolic approaches.

Centralized approaches are normally combined with centralized SHE architectures. They usually involve intelligent algorithms and methods (combining the symbolic and subsymbolic approach), including fuzzy logic, neural networks, clustering, pattern mining, Markov decision processes. Distributed approaches include the utilization of multi-agent systems combined with ontologies and rule-based reasoning (symbolic approaches).

In the abnormal usage detection based on sequence patterns, there have been many proposed methods for detection abnormal usages of appliances in a smart home environment, such as in [5–7]. Moreover, Neural network approach has been done to predict the future values which are used to inform the caregiver in case anomalous behavior is predicted in [8, 9]. Besides, Cook et al. [7, 10, 11] proposed several frameworks to mine energy data and extend a suffix tree data structure and then use a clustering algorithm to detect energy patterns outliers which are far from their cluster centroids. Juan et al. [12] propose a technique that integrates the semantics of sensor readings with statistical outlier detection. Moreover, conceptual studies and used cases reported for abnormal events in the smart home context were proposed by some authors in [13]. An activity recognition system using for dementia care uses Markov Logic Network approach to detect abnormality in occupant behavior was presented in [14]. Huang et al. [15] proposed a method to recognize abnormal habits using duration histogram and information provided by intelligent space. Other researcher in [16] proposed a three-level hierarchical optimization approach to solve scalability, computational overhead, and minimize daily electricity cost.

Previous researches of abnormal detection mainly focused on sequence pattern [8] and probability density function based on EVT [5, 17–19]. Furthermore, a heuristic novelty threshold has set on the pdf $f(x) = k$, such that x is abnormal when $f(x) < k$. $f(x)$ is used simply as abnormal score, and the threshold is set such that separation between normal and abnormal data is maximized on the validation dataset [20]. Some other approaches [19] use the cumulative probability F_n associated with f_n . They compute the probability mass obtained by integrating f_n over the region R where f_n exceeds the novelty threshold.

However, previous works focused on analysis of the usage behavior on single device and neglect of the appliance correlations.

2.4 SHE Applications

There is no limit for SHE applications, being restricted just by the human creative. Based on our literature survey, we have distinguished four main application areas of SHE, in particular [3]:

- Elderly/Aging/Home Care
- Energy Efficiency
- Comfort/Entertainment
- Safety/Security.

The areas are not essentially separate. For instance, security can be connected with maturing and older folks. In addition, capacities having a place with one or more distinctive application types can be found inside of the same SHE. Finally, these applications can offer computational methodologies. For example applications identified with alders and wellbeing all the time use computational techniques for video surveillance [21].

3 Proposed Methods

In this section, we propose a system framework for Abnormal Usage Detection in smart home environments as in Fig. 2. First, we collected the usage data of all appliances by smart meters and sent the data log to a server. After that we transform the data into correlation patterns using CoPMiner algorithm [1]. Then our system uses the correlation patterns to detect abnormal usage behavior. Finally, we output all abnormal extraordinary behaviors to users. We also present three methods for abnormal usage detection.

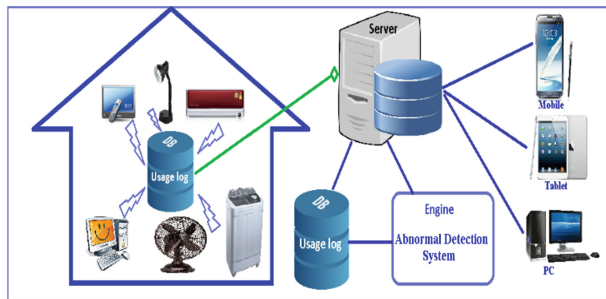


Fig. 2. A framework for abnormal usage detection system.

Before going to the methods, two definitions are provided.

Definition 1. Sub pattern at a time period and sub-patterns set at a time period: Given a correlation pattern P, a correlation sub-pattern at a time period of two appliances is a subset of correlation pattern P if end-time of the first appliance occurs after start-time of the second appliance. A set of sub-patterns at a time period is the collection of all sub-patterns at the time of all correlation patterns.

We take the database in Table 1 as an example. Let P1 be [A+|B+|C+|C-|A-|B-], a sub-pattern is [A+], a set of sub-patterns at a time period is S1 = {[A+], [A+B+]},

Table 1. An example of a correlation pattern set, each frequent sequence has two part: the order of appliances' occurrence and number of occurrence's times

Frequent sequences	
[A+ A-]: 65	[B+ B- C+ C-]: 4
[A+ B+ A- B-]: 12	[C+ C-]: 67
[A+ B+ C+ C- A- B-]:2	[C+ C- C+ C- C+ C-]:3
[A+ B+ D+ B- A- D-]:2	[C+ C- C+ C- D+ D-]:22
[B+ B-]: 66	[D+ D-]:55
[B+ B- C+ A+ A- C-]:5	[D+ D- C+ C-]:25
[B+ B- C+ C-]: 20	[D+ D- C+ D+ D- C-]:6
[B+ B- C+ C- C+ C-]:6	[D+ D- D+ A+ A- D-]:2
Probability density functions:	
$f_{A^+} \cdot f_{A^-} \cdot f_{B^+} \cdot f_{B^-} \cdot f_{C^+} \cdot f_{C^-} \cdot f_{D^+} \cdot f_{D^-}$	

[A+B+C+]]. Let P2 be [B+|B-|C+|A+|A-|C-], a set of sub-patterns at a time period is $S2 = \{[B+], [C+], [C+A+]\}$.

Definition 2. Appliances' Combinations. Given a pattern P, a combination is a way of selecting appliances from P, such that the order of selection does not matter. A subset S is a combination of appliances in P, denoted by S P.

For example, P is [A+B+C+], there are seven subsets $S = \{[A+], [B+], [C+], [A+B+], [A+C+], [B+C+], [A+B+C+]\}$. A list of sub-patterns is a combination of all appliances with 2 k-1 subsets. Notice that we do not use empty sub pattern.

In the following sections we review three methods for abnormal usage detection based on our previous work [22].

3.1 Method 1: Calculating Anomaly Score Using Extreme Value Theory

Anomaly detection using EVT approach is based on a model of normal behavior which was presented by the probability distribution function.

Given a data set D (correlation pattern set), consisting of appliances, each appliance has the probability density function (pdf) $y = f(x)$ which build from its training dataset. Anomaly detection address the question whether a query pattern $Q = \{q1, \dots, qk\}$ is drawn from $f(x)$ or not. Each appliance is Q has the corresponding density values based on pdf $f(x)$.

$$(y1, \dots, yk) = (f(q1), \dots, f(qk))$$

First, based on the Eq. (1), we compute a set of y_{min} in which the distribution is lower than the threshold. The threshold will be set based on the size of training dataset. If a training dataset has m events then setting threshold at $size = \ln(m)$ Then, set $F(x) = size/100$, the $F(x)$ is given as:

$$F(x) = \int_R f_n(x)dx = size\%$$

The set $y_{min} = \{y|x \in D \text{ and } F(x) = size/100\}$: The distribution of y_{min} describes the distribution of minima of training dataset. An anomaly may be located in the tails of pdf f or between the modes of f . We find y_{min} which is the tails of f or the low probability between the modes of f .

The next step is that we apply the Weibull distribution for y_{min} . The form of the 3-parameter Weibull distribution is commonly used in practice

$$w(y_{min}) = \frac{\beta}{\eta} \left(\frac{y_{min} - \gamma}{\eta}\right)^{\beta-1} \exp\left(-\left(\frac{y_{min} - \gamma}{\eta}\right)^\beta\right)$$

Where parameters $\beta > 0$, η and γ are shape, scale, and location respectively. The location parameter, γ , locates the distribution along the abscissa. The distribution moves to the right (if $\gamma > 0$) or to the left (if $\gamma < 0$). We set $\gamma = 0$, the distribution starts at the origin. The parameters β and η can be found by using maximum likelihood estimates. The 2-parameter Weibull is obtained by setting $\gamma = 0$, and is given by

$$w(y_{min}) = \frac{\beta}{\eta} \left(\frac{y}{\eta}\right)^{\beta-1} \exp\left(-\left(\frac{y}{\eta}\right)^\beta\right)$$

Since the probability of these are likely to very close to zero, the use of log helps emphasize their differences. The transformation is given as:

$$t = -\log(w)$$

Using this transformation, the short tail near zero of the Weibull distribution is then stretched out as the right tail of the Gumbel distribution for maxima. Hence, extreme values can be shown more clearly. The cumulative distribution function of the Gumbel distribution is

$$G(t) = \exp\left(-\exp\left(-\frac{t-c}{d}\right)\right)$$

where $c = 1/\beta$ and $d = -\ln(\eta)$.

In abnormal detection, extrema are regarded as potentially anomaly. The final step is that we define an anomaly score for each appliance. Hence, we can compute the anomaly score for each appliance is query pattern Q . Anomaly score, AS1, can be defined as:

$$AS1 = G(t) \tag{1}$$

Note that AS1 takes low values if x is close to the center of the distribution and increases as x becomes more abnormal (Fig. 3).

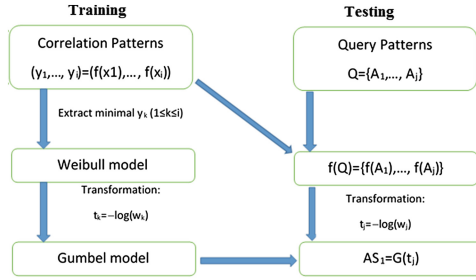


Fig. 3. Schematic of the algorithm

3.2 Method 2: Calculating Anomaly Score Based on Sequence Patterns

We explore the frequent sequences of correlation patterns dataset to calculate the anomaly score for each appliance. Table 1 shows the set of frequent sequences which can be used to determine abnormal events. We assume that all sequence patterns in this dataset are normal patterns. All appliances occurred in normal scenarios. For example, a pattern, $[A+|B+|D+|D-|B-|E+|E-|A-]$ describes a normal occurrence order of four appliances (A, B, D, and E). We can extract this sequence into four sub patterns as $[A+]$, $[A+B+]$, $[A+B+D+]$, and $[A+E+]$ as definition 1. This means that $[A+]$ can turn on while $[D+]$ turns on but $[D+]$ and $[E+]$ cannot turn on at the same time. This is necessary to decompose a correlation pattern into sub patterns because an appliance can occur many times and it is difficult to take out appliances which occur at a time period.

Algorithm 1: Method2(CP, Q)
Input: a correlation pattern dataset CP, a query pattern Q.
Output: all abnormal appliances A.
01: $A \leftarrow \phi$;
02: transform CP into sub pattern L by Definition 1;
03: transform Q into a combination list QL by Definition 2;
04: evaluate proportion $P(x)$ for each element x in QL;
05: compute anomaly score AS2 for each appliance;
06: $A \leftarrow \min(AS2)$;
07: output all appliances in A;

For unknown pattern, we actually do not know the order of appliances in this pattern. The order in query pattern is random, which depends on users' input. Therefore, it is not easy for us to compare the query pattern with existing correlation pattern set. Our method use probability theory to solve this problem.

Given a dataset D (correlation pattern set), consisting of appliances and their occurrences in classical sense, denoted by Ω . It is then assumed that for each element $x \in \Omega$.

For the unknown pattern $Q = \{X1, X1, X3\}$ of space Ω , assume that all appliances in Q have occurred in dataset D. Anomaly score of Xi is defined as:

$$AS_2(X_1) = f(X_1) + f(X_1X_2) + f(X_1X_2) + f(X_1X_2X_3)$$

$$f(X_1) = \frac{\text{number of occurrences}}{\text{total of sub patterns}}$$

where

$$f(X_1X_2) = \frac{\text{number of occurrences of } X_1 \& X_2 \text{ together}}{\text{total of sub patterns}}$$

$$f(X_1X_3) = \frac{\text{number of occurrences of } X_1 \& X_3 \text{ together}}{\text{total of sub patterns}}$$

$$f(X_1X_2X_3) = \frac{\text{number of occurrences of } X_1, X_2 \& X_3 \text{ together}}{\text{total of sub patterns}}$$

We also define some rules helps us identify the anomalies.

The rules are given as:

If $P(Q) > 0$ then the query pattern Q has no abnormal usage behavior.

If $P(Q) = 0$ then calculate $AS_2(X_i)$ for each appliance.

$\text{Min}(AS_2(X_i))$ is abnormal.

We need to decompose correlation patterns into sub patterns because of two reasons. First, we want to combine appliances which may turn on at a time period while an appliance can turn on and turn off many times in a correlation pattern. Each sub pattern indicates that appliances may turn on in the same period, which can be used for computing appliance anomaly scores. Second, we do not use turned off information in correlation patterns for our method because we have no turned off values of appliances in the query sequence. Therefore, we can eliminate turned off symbol (-) in sub patterns.

The first advantage of the second method is that we do not need time information of appliances to determine abnormal usage behavior since there is no time information in frequent sequences of correlation patterns. Another advantage of this method is that we do not pay attention to the order of appliance when compute anomaly scores because we have no order information in query pattern. However, we cannot identify exactly abnormal usage behaviors when there is only one appliance in the query pattern. If there are many appliances with low scores, we can use a threshold instead of using the minimum value.

3.3 Method 3: Determined Abnormal Time Intervals

First, we define time intervals of occurrence of each appliance in the correlation pattern that is to identify appropriate time periods for each appliance.

One possible approach is to generate the intervals while the correlation patterns discovery part. However, the time complexity increases, since all possible intervals have to be considered. The potential drawback is that the quality of accuracy can be

affected by how we define the intervals. However, we can minimize this possibility if we do not fixed-width time intervals. Instead determine the intervals based on the original dataset, we use the probability density function. The area under the curve from time t_1 to time t_2 gives us the probability that an appliance will turn on between t_1 and t_2 .

We need to define the time intervals as intervals between local maxima of the probability density function. The main idea behind this approach is that a user often turn-on this appliance during a certain time period. For example, as illustrated in Table 3, a user may usually turn-on in two periods as between 03:00 AM and 06:30 AM; between 17:30 PM and 21:00 PM. Since some appliances turn on more frequently than others, we define the time intervals by computing probability density functions for each appliance separately.

<p>Algorithm 2: Method3(pdf, Q)</p> <p>Input: probability density function set pdf, a query pattern Q.</p> <p>Output: all abnormal appliances A.</p> <p>For each appliance:</p> <p>$\forall t_i, t_j \in M :$</p> <p>Let x_1 be $t_i \in M$ such that $\begin{cases} f(t_i) \geq k \\ f(t_{i-1}) < k \end{cases}$ ort $t_{i-1} \notin M$</p> <p>Let x_2 be $t_j \in M$ such that $\begin{cases} f(t_j) \geq k \\ f(t_{j+1}) < k \end{cases}$ ort $t_{j+1} \notin M$</p> <p style="text-align: center;">$interval_k \leftarrow [x_1, x_2]$</p> <p>$A \leftarrow appliance \in interval_k$</p> <p>Output all appliance in A.</p>

For anomaly detection, an appliance in the query pattern will be determined whether it is normal or abnormal. The appliance is normal when its time is in this appliance's time intervals. Each appliance will be determined by its time intervals. All appliances in query pattern must be existed in dataset.

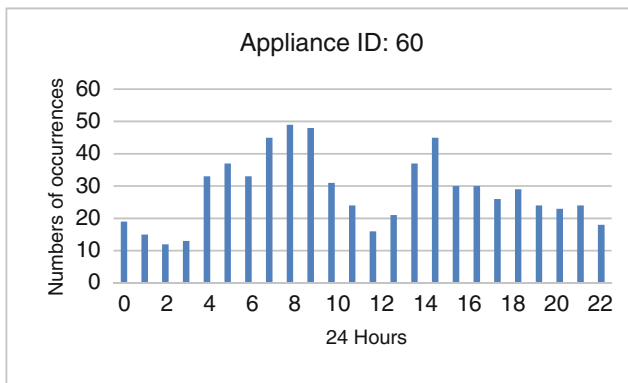
4 Experimental Results

With our framework, we can easily collect energy consumption data of appliances of the houses. However, we have not employed smart meters to collect usage data log of appliances in a real home. Hence, this performance study has been conducted on a real world dataset which is taken from [1] including six appliances as presented in Table 2. First, we implement our three methods on data set in detail. Second, we compare the execution time using real world dataset at different threshold size. Finally, we compare the accuracy of the three methods on real world dataset.

Table 2. Data analyzed – correlation pattern sets

Appliance ID	Appliance name	Observations
13	Kitchen outlet – 1500 W	505
17	Kitchen outlet – 30 W	624
24	Washer dryer 3 W	497
29	Outlets	238
57	Furnace	258
60	Smoke alarm	682

Figure 4 shows the plot of the $n = 682$ observed of the appliance ID 60. This is multimodal distributed such that anomalies possibly occur between the modes. Hence, we not only consider the left and the right tail of the distribution but also between the modes. We use maximum likelihood estimation, which is one of the most common estimation procedures used in practice. We also compute likelihood based interval estimates of the parameters and the quantities of interest which provide additional information related to the accuracy of the point estimates. These intervals, contrarily to those based on standard errors, do not rely on asymptotic theory results and restrictive assumptions. We expect them to be more accurate in the case of small sample size. Another advantage of the likelihood-based approach is the possibility to construct joint confidence intervals. The greater computational complexity of the likelihood-based approach is nowadays no longer an obstacle for its use.

**Fig. 4.** Numbers of occurrences per hour for 45 days.

The implementation of the proposed method involves the following steps: select the threshold u , fit the Weibull and Gumbel distributions and then compute anomaly score.

(a) Selection of the threshold u

We know that the higher the threshold the less observations are left for the estimation of the parameters of the tail distribution function. There is no automatic algorithm with satisfactory performance for the selection of the threshold u . In

previous work [23, 24] the threshold has been set on the probability density function (PDF) $f_n(x) = k$. In this work, we define the threshold for cumulative distribution function (CDF) $F(x) = \text{size}$. The size value is based on the size of samples dataset. Appliance 60 has 682 samples observations, we have $\ln(682) = 6.525$. Hence, we collect the set y_k such that $F(k) = 6.525\%$. The number of observations exceeding the threshold is 53.

(b) Maximum likelihood estimation

Given the theoretical results presented in [23], the distribution of the observations that we collect above should be drawn a Weibull distribution. We compute the value β and η that maximize the log-likelihood function for the sample y_k .

(c) Weibull and Gumbel distributions

An anomaly score of an appliance is defined as (1). An appliance which its anomaly score is high and between the Gumbel value of $f(x)$ can be viewed as anomaly. In other words, an appliance is abnormal if this pdf value is lower than a threshold.

Table 3. The PDF value and Gumbel value (anomaly score) of appliance 60 from 00:00 am to 23:30 pm

Time	pdf	AS ₁	Time	pdf	AS ₁
00:00	4.421	0.9737	12:00	3.418	0.9483
00:30	4.068	0.9666	12:30	3.49	0.9508
01:00	3.887	0.9623	13:00	3.483	0.9505
01:30	3.835	0.9609	13:30	3.373	0.9467
02:00	3.853	0.9614	14:00	3.192	0.9399
02:30	3.866	0.9618	14:30	3.014	0.9323
03:00	3.799	0.96	15:00	2.897	0.9269
03:30	3.625	0.955	15:30	2.864	0.9253
04:00	3.399	0.9477	16:00	2.9	0.927
04:30	3.197	0.9401	16:30	2.966	0.9302
05:00	3.057	0.9343	17:00	3.036	0.9334
05:30	2.983	0.9309	17:30	3.108	0.9365
06:00	2.949	0.9294	18:00	3.182	0.9395
06:30	2.918	0.9279	18:30	3.243	0.9419
07:00	2.866	0.9254	19:00	3.273	0.9431
07:30	2.797	0.9219	19:30	3.284	0.9435
08:00	2.735	0.9186	20:00	3.297	0.944
08:30	2.7	0.9168	20:30	3.319	0.9448
09:00	2.703	0.9169	21:00	3.336	0.9454
09:30	2.752	0.9196	21:30	3.338	0.9455
10:00	2.852	0.9247	22:00	3.348	0.9459
10:30	2.994	0.9315	22:30	3.403	0.9478
11:00	3.154	0.9384	23:00	3.539	0.9524
11:30	3.302	0.9441	23:30	3.791	0.9598

Table 3 illustrates the pdf and Gumbel information of appliance 60 in a day. We can see that anomaly scores are higher than that of other scores when appliance turns on between 23:00 and 03:30. From a probabilistic point of view, a typical choice of threshold is 95 %. Appliance 60 occurrences will be abnormal when anomaly score is greater than 0.95.

In the following experiments, we compare the running time of Method 1 and Method 3 method with threshold varied from 5 % to 10 % on A17-N624 dataset, while Method 2 test with query patterns varied length from 1 appliance to 6 appliances. A17-N624 dataset contains 624 events of appliance 17. Figure 5 shows the running time of the three methods with different threshold and number of appliances in the query patterns. Obviously, when we continue to higher the threshold, the runtime for Method 1 and Method 3 remain unchanged at around 56 (seconds). We can see that when the number of appliances increases, the processing time required for Method 2 increases. This is partly because Method 1 and Method 3 use the probability density function information while Method 2 uses frequent sequences dataset. Many appliances in a query pattern lead to generate more number of combination sequences.

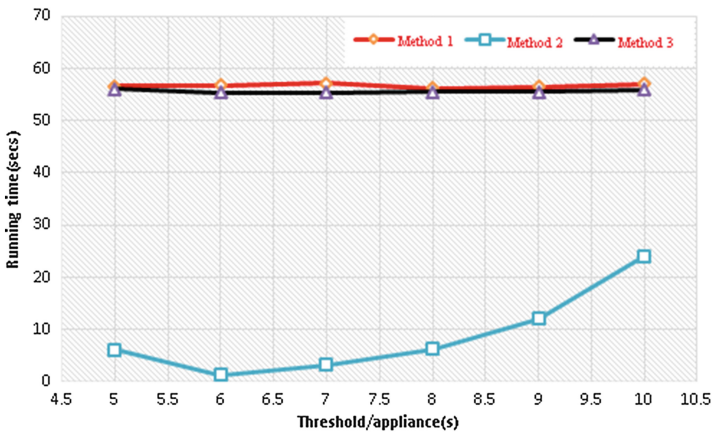


Fig. 5. Running time testing on A17-N624 dataset.

For testing the performances of anomaly detection, we have to generate the synthetic query because the original data does not label every day with normal or abnormal behavior. We take 1440 queries (1440 min per day), $Q = \{t_1, t_2, \dots, t_i | t_i \in [0, 24]\}$.

Figure 6 shows the percentage of accuracy for each method. We can see that using probability density function information outperform taking frequent sequences. Method 1 (precise at 85 %) and Method 2 take more precise than that of Method 2, thus, these methods would be promising for abnormal usage detection.

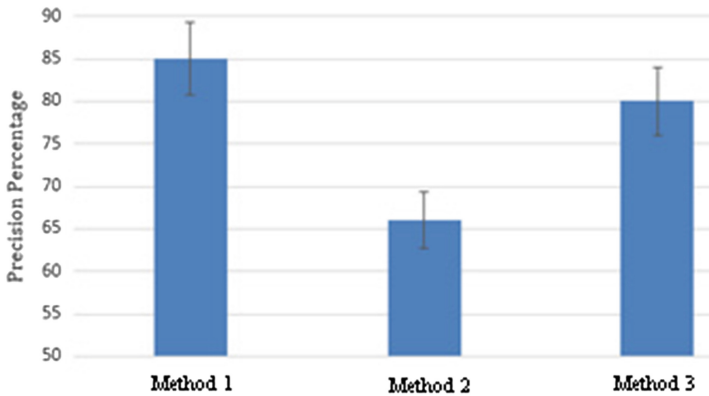


Fig. 6. Precision testing on real world dataset.

5 Conclusion

In this work, we review some definitions and related problems of the Smart Home Environments (SHEs) such as their classification, architecture, methods, and applications to provide the reader an overview on the areas of SHEs. Moreover, we propose a framework for abnormal usage detection system which can be used to detect abnormal usage behaviors in SHEs using home appliance usage logs.

Experimental results on real world data set indicate that the proposed framework can be useful for detection and notify abnormal behaviors to the users. We will continue to improve the proposed methods in the future.

References

1. Chen, Y.-C., Chen, C.-C., Peng, W.-C., Lee, W.-C.: Mining correlation patterns among appliances in smart home environment. In: Tseng, V.S., Ho, T.B., Zhou, Z.-H., Chen, A.L., Kao, Hung-Yu. (eds.) PAKDD 2014, Part II. LNCS, vol. 8444, pp. 222–233. Springer, Heidelberg (2014)
2. Chen, Y.-C., Ko, Y.-L., Peng, W.-C., Lee, W.-C.: Mining appliance usage patterns in smart home environment. In: Pei, J., Tseng, V.S., Cao, L., Motoda, H., Xu, G. (eds.) PAKDD 2013, Part I. LNCS, vol. 7818, pp. 99–110. Springer, Heidelberg (2013)
3. Badica, C., Brezovan, M., Badica, A.: An overview of smart home environments: architectures, technologies and applications. In: Georgiadis, C.K., Kefalas, P., Stamatis, D. (eds.) BCI (Local). CEUR-WS.org, vol. 1036, p. 78 (2013)
4. Alam, M.R., Reaz, M.B.I., Ali, M.A.M.: A review of smart homes—past, present, and future. *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.* **42**, 1190–1203 (2012)
5. Luca, S., Karsmakers, P., Cuppens, K., Croonenborghs, T., Vel, A.V.D., Ceulemans, B., Lagae, L., Huffel, S.V., Vanrumste, B.: Detecting rare events using extreme value statistics applied to epileptic convulsions in children. *Artif. Intell. Med.* **60**, 89–96 (2014)
6. Jakkula, V., Cook, D.J., Crandall, A.S.: Temporal pattern discovery for anomaly detection in a smart home. In: 3rd IET International Conference on Intelligent Environments, 2007. *IE 07*, pp. 339–345 (2007)

7. Jakkula, V., Cook, D.J.: Detecting anomalous sensor events in smart home data for enhancing the living experience. *Artif. Intell. Smarter Living*, WS-11-07. AAAI (2011)
8. Ovidiu Aritoni, V.N.: A methodology for household appliances behaviour recognition in AmI systems. In: *The Seventh International Conference on Autonomic and Autonomous Systems*, ICAS 2011 (2011)
9. Lotfi, A., Langensiepen, C., Mahmoud, S., Akhlaghinia, M.J.: Smart homes for the elderly dementia sufferers: identification and prediction of abnormal behaviour. *J. Ambient Intell. Hum. Comput.* **3**, 205–218 (2012)
10. Minor, B., Cook, D.J.: Regression tree classification for activity prediction in smart homes. In: *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*, pp. 441–450. ACM, Seattle, Washington (2014)
11. Rashidi, P., Cook, D.J.: COM: a method for mining and monitoring human activity patterns in home-based health monitoring systems. *ACM Trans. Intell. Syst. Technol.* **4**, 1–20 (2013)
12. Juan, Y., Stevenson, G., Dobson, S.: Fault detection for binary sensors in smart home environments. In: *2015 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pp. 20–28
13. Tran, A.C., Marsland, S., Dietrich, J., Guesgen, H.W., Lyons, P.: Use cases for abnormal behaviour detection in smart homes. In: Lee, Y., Bien, Z., Mokhtari, M., Kim, J.T., Park, M., Kim, J., Lee, H., Khalil, I. (eds.) *ICOST 2010*. LNCS, vol. 6159, pp. 144–151. Springer, Heidelberg (2010)
14. Gayathri, K.S., Elias, S., Ravindran, B.: Hierarchical activity recognition for dementia care using Markov Logic Network. *Pers. Ubiquitous Comput.* **19**, 271–285 (2015)
15. Huang, B., Tian, G., Wu, H., Zhou, F.: A method of abnormal habits recognition in intelligent space. *Eng. Appl. Artif. Intell.* **29**, 125–133 (2014)
16. Jarrah, M., Jaradat, M., Jararweh, Y., Al-Ayyoub, M., Bouselham, A.: A hierarchical optimization model for energy data flow in smart grid power systems. *Inf. Syst.* **53**, 190–200 (2015)
17. Andreeva, V.O., Tinykov, S.E., Ovchinnikovaa, O.P., Parahinc, G.P.: Extreme value theory and peaks over threshold model in the Russian stock market. *J. Siberian Fed. Univ. Eng. Technol.* **1**, 111–121 (2012)
18. Clifton, D., Huguency, S., Tarassenko, L.: Novelty detection with multivariate extreme value statistics. *J. Sign. Process. Syst.* **65**, 371–389 (2011)
19. Smith, M., Reece, S., Roberts, S., Rezek, I.: Online maritime abnormality detection using gaussian processes and extreme value theory. In: *2012 IEEE 12th International Conference on Data Mining (ICDM)*, pp. 645–654 (2012)
20. Nairac, A., Corbett-Clark, T.A., Ripley, R., Townsend, N.W., Tarassenko, L.: Choosing an appropriate model for novelty detection. In: *Fifth International Conference on Artificial Neural Networks (Conf. Publ. No. 440)*, pp. 117–122 (1997)
21. Brezovan, M., Badica, C.: A review on vision surveillance techniques in smart home environments. In: *2013 19th International Conference on Control Systems and Computer Science (CSCS)*, pp. 471–478 (2013)
22. Minh-Thai, T.N., Thai-Nghe, N.: Methods for abnormal usage detection in developing intelligent systems for smart homes. In: *Proceedings of the Seventh International Conference on Knowledge and Systems Engineering (KSE 2015)*, pp. 1–6. IEEE Xplore (2015)
23. Nagatsuka, H., Kamakura, T., Balakrishnan, N.: A consistent method of estimation for the three-parameter Weibull distribution. *Comput. Stat. Data Anal.* **58**, 210–226 (2013)
24. Lauer, M.: A mixture approach to novelty detection using training data with outliers. In: Flach, P.A., De Raedt, L. (eds.) *ECML 2001*. LNCS (LNAI), vol. 2167, pp. 300–311. Springer, Heidelberg (2001)

Emerging Data Management Systems and Applications

Modelling Sensible Business Processes

David Simões¹, Nguyen Hoang Thuan², Lalitha Jonnavithula²,
and Pedro Antunes²(✉)

¹ Faculty of Sciences, University of Lisbon, Lisbon, Portugal
david.simoese@esce.ips.pt

² School of Information Management, Victoria University of Wellington,
Wellington, New Zealand
{Nguyen.Thuan, Lalitha.Jonnavithula,
Pedro.Antunes}@vuw.ac.nz

Abstract. In this paper we develop the concept of sensible business process, which appears in opposition to the more traditional concept of mechanistic business process that is currently supported by most business process modelling languages and tools. A sensible business process is founded on a rich model and affords predominant human control. Having developed a modelling tool supporting this concept, in this paper we report on a set of experiments with the tool. The obtained results show that sensible business processes (1) capture richer information about business processes; (2) contribute to knowledge sharing in organisations; and (3) support better process models.

Keywords: Sensible processes · Process stories · Business process modelling · Business process management

1 Introduction

Business Process Management (BPM) has evolved towards a mature discipline concerned with the transformation of business goals, rules, processes, and practices into electronic services. Built on top of a variety of enterprise software and infrastructural components such as workflow engines, enterprise resource planning, service-oriented architectures and information repositories, BPM has provided broad facilities to manage business processes, which potentially increase productivity and reduce cost [1]. The typical BPM lifecycle includes eliciting and analysing process-related information, designing process models using specialised tools and languages, enacting process rules in enterprise systems, and executing/maintaining the services [2].

According to this lifecycle, the success of a BPM initiative starts with good elicitation, analysis and design, so that when reaching the enactment stage, the electronic services will effectively deliver the envisaged business goals. Of course, ensuring success is relatively easy in the case of *purely automated systems*, since their scope is well delimited, workflows are known, and procedures are always applicable. In these systems, systematic and preventive verifications of the relationships between process models and actual data processing usually ensure that services can be continuously provided within the required service-level agreements. Furthermore, exceptions in

purely automated organisations tend to be expected exceptions, which can also be handled by pre-programmed instructions [3, 4].

Though the situation becomes much more challenging in areas where service provision involves a mix between humans and machines. Example areas include healthcare and customer relationship management, where human discretion is often necessary to resolve unique business cases [5]. In these areas, BPM needs to coordinate human decisions and automatic processes, which challenges the concept of purely automated system. Underlying these challenges, we find the different capabilities and constraints of humans and machines, e.g., machines can process more symbolic information in parallel and humans have more capacity for processing perceptual information [6]. Furthermore, humans have more capacity for recognising and interpreting context, making decisions with information gaps, and accommodating and improvising [7, 8].

Additionally, the BPM discipline must consider a business reality characterised by ever changing business contexts and goals, diverse clients' needs, unexpected events, and emergent human behaviour. In such scenario, BPM experts may have to carefully consider the risks and consequences of mismatched process models and enacted operations, a problem that has been generally coined the "model reality divide" [9, 10], which is ultimately related with other problems predating BPM technology like the "lack of realism" (when rules do not exactly apply to the situation), "lack of details" (when precise rules about the situation are missing), and "lost in translation" (when rules have been erroneously converted to machine language) [11]. All these problems underline how difficult it is to integrate human and automated behaviour.

The BPM discipline has its roots in software engineering and computer science. Formal theory and methods such as Petri Nets, Pi-Calculus, and the Entity-Relationship and Relational models have been widely used to model data and processes [1]. Standards such as BPMN [12], UML [13], IDEF0 [14], BPEL [15], XPDL [16], and BPQL [17], just to mention few, have been developed to help specifying business processes and process-related data in consistent and valid ways. Besides, an extensive body of research literature has been published concerning the requirements and constraints imposed by process enactment and execution. The concerned topics include avoiding deadlocks and live-locks, allowing model/language transformations, and avoiding inconsistent system states, system failures, unreachable states, racing conditions, non-determinism, data integrity failures, etc. [3, 18].

We argue that these concerns reflect a mechanistic view of the BPM approach. While the success of current BPM technology is beyond any doubt, there has been some recent concern on several shortcomings, biases, omissions, and problems this approach has. Among these concerns we find, for instance, the lack of implicitness [10], struggle for flexibility [19, 20], and lack of consideration for tacit knowledge [21]. Overall, these problems suggest that perhaps a more sensible viewpoint of this technology is needed. A sensible perspective privileges the integration of human knowledge, context-awareness, diversity, creativity, ambiguity, and many other properties pertaining to human behaviour in BPM systems [22]. This viewpoint leads to *sensible business processes*, which balance the level of control between machine and human within the BPM systems.

In the next section we elaborate our definition of sensible business process. Section 3 discusses results from three experiments assessing the elicitation and design of sensible business process models. In Sect. 4 we discuss the results and provide some implications for research.

2 Sensible Versus Mechanistic BPM

In the introduction we argued that the BPM lifecycle considering eliciting, analysing, modelling, and enacting business processes in organisations has been significantly constrained by the final stage and in particular the translation of process models into machine-readable instructions. The focus on a more sensible perspective, where the BPM practice may be less constrained by technology, suggests we should consider the issue of control in technology support.

In Fig. 1 we illustrate that the level of control over a human-machine system is a combination of two variables: human and machine control. The type of supporting technology determines such combination. Technology may either enforce strict rules and procedures over human activities or support open-ended, unrestricted human activities. In between, we find what has been designated as joint-cognitive systems, where control is a co-agency between humans and machines [6]. According to the joint-cognitive perspective, details of the real world may determine a swift change of control between the two parties, either because the machine may try to compensate for human error, or the other way around. For BPM in general and process enactment in particular, this means that enterprise systems should be designed for different levels of flexibility required by the work environment [20].

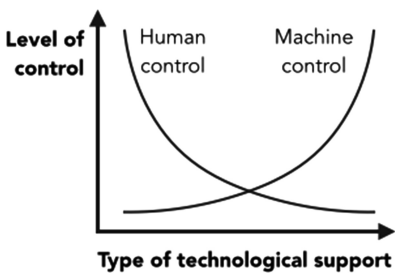


Fig. 1. Level of control (adapted from [23])

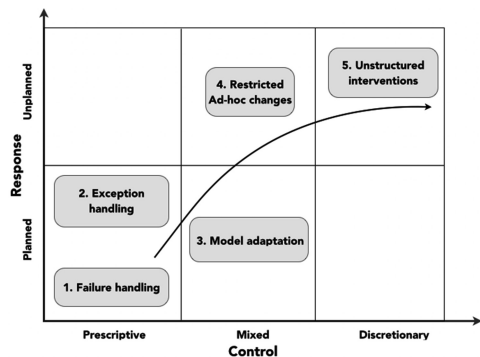


Fig. 2. Exception handling strategies (adapted from [3])

The joint support to human and machine control has significant implications for enterprise systems, especially regarding the implementation of exception handling mechanisms. In prior research we identified five types of exception handling

mechanisms, which can be conceptualised in two dimensions considering the type of control and type of response [3, 24].

We defined three types of control (Fig. 2): prescriptive, where machines apply predefined handling procedures and therefore human intervention is highly constrained; discretionary, where humans take control and decide what to do next; and mixed, considering situations where control has to be negotiated between humans and machines. We defined two types of response, which may be either planned or unplanned. In the planned case, humans and machines have predefined exception-handling procedures, which can therefore be applied to resolve an exception, while in the unplanned case, no procedure is available and the handling procedure has to rely on other strategies, usually involving human ingenuity.

Using these two dimensions, we can now characterise the five exception-handling strategies. They may range from low-level, automated failure handling (e.g. wait for the network to recover from failure), to high-level, programmed exception handling (e.g. rollback a transaction in case of message failure), model adaptation (e.g. change the flow and conditions, if they do not impact other processes), restricted ad-hoc changes (e.g. add an activity between two consecutive activities), and unrestricted interventions (e.g. add or delete activities without consideration for model consistency).

Besides the problem of control, we should also discuss the differences between process models and business reality. By definition, any business process model is always an incomplete representation of the business reality [25]. However, we argue that here again we may consider that the level of modelling is a combination of two variables: contextualisation and normalisation. In Fig. 3, we use the concept of level of modelling to characterise how a process model may reflect the work reality by either leaning towards the normalisation or towards the contextualisation of work. On the one hand, normalisation seeks to find a single process model describing the regular/consensual sequence of activities, eventually with a great level of detail. On the other hand, contextualisation considers the large number of possible variations in process execution. Of course once again these two different approaches to modelling may require different types of support from enterprise systems.

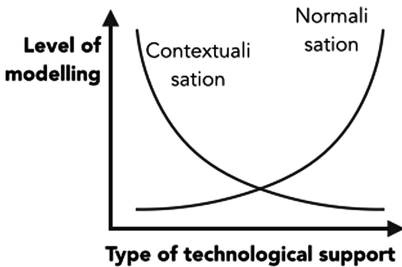


Fig. 3. Level of modelling

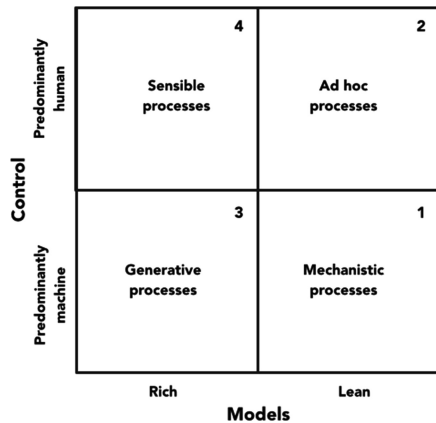


Fig. 4. Classification of processes

With these two dimensions of the problem, considering level of control and level of modelling, we may now discuss with more detail what types of processes fall in each category (Fig. 4). We first note that quadrant 1, favouring the normalisation of work with predominant machine control, is the domain of mechanistic BPM processes. They favour behavioural clarity and predictability. They avoid disturbances and human decision-making. With these characteristics, the processes are strongly suitable to mechanisation and computerisation [6].

Quadrant 2 suggests the support to ad hoc processes, where the dynamic flow of events, including unexpected events, determines the process evolution [24]. Health care treatments are typical examples of ad hoc processes, which usually deal with exceptional or unanticipated situations [26]. This kind of processes is characterised by significant human intervention in sensemaking the situation and decision-making. Here, an important role attributed to machines is to support the decision makers e.g. with visualisation tools, query and filtering mechanisms, etc. [24].

The combination of predominant machine control with rich models fosters the machines' capacity to generate and handle an infinite number of alternatives (quadrant 3). Research in generative design highlights how technology may inspire alternative solutions through evolution, breeding and adaptation [27]. According to this perspective, variety is not only possible but also desirable and exceptions, instead of representing a setback when analysing, modelling and managing business processes, may actually become an opportunity to improve a business process.

The combination of predominant human control and rich process modelling concerns sensible processes (quadrant 4). Here, human sensemaking and decision-making capabilities combine with rich information necessary to adapt the process to the changing environment both through human decision-making and through computational support [28, 29]. Management and governance literature has provided several instances of sensible processes. For example, Pries-Heje and Baskerville [30], while examining the process of organisational change, identified many ways to enact organisational changes, all based on sound competing theories. This situation strongly requires human capabilities to analyse rich organisational information and to make sensible choices, which can be supported by computational tools.

The current paper focuses exactly on this quadrant. We characterise sensible business processes as *processes that leverage both the human capacities for decision-making and the information processing capacities for supporting the sense-making process*. Thus sensible business processes appear in opposition to mechanistic processes, where modelling and control predominantly rely upon and utilise the machines' capacities. In the next section we describe research addressing the elicitation and modelling of sensible business processes.

3 Eliciting and Modelling Process Stories

The rich-lean perspective suggests that business process models may capture richer information on how work is done in organisations. This led us to develop an exploratory research agenda centred on the following research questions.

- What process knowledge would be captured?
- What methods and tools would be needed to capture such knowledge?
- What would be the effectiveness of these tools and methods?

In [10, 31] we discussed the first two questions. Building on prior research [32], we suggested modelling “process stories”. A process story is a diverse collection of structured and unstructured information about a business process, which may integrate different perspectives and various narrative elements. Its conceptual foundation lies in Organisational Storytelling theory [33]. According to Denning [33], stories communicate complex ideas and spring people into action using narrative mechanisms. Furthermore, stories bring detailed explanations, contextual information, values, and what-if considerations to knowledge sharing.

Based on the storytelling theoretical foundations, we developed an information model for process stories [10]. A process story has a beginning and ending, descriptive attributes, triggers, and a sequence of scenes. The critical element to structure a process story is the scene. It combines visual with textual information to describe a work setting, presenting the actors, suggesting the social atmosphere, and explaining what happens in terms of events and action. A scene contains an abstract picture, which could also be described as a cartoon, of a business situation such as checking a form, contacting a client, having a meeting, and signing a document. By associating pictures to scenes we allow business people to analyse a process story by recognition and familiarity with the depicted situations.

Besides the abstract picture, a scene contains semi-structured information about actors, artefacts, events, and actions, which may be involved in the depicted work situation. Dialogue lines may also be associated to actors appearing in a picture, which follows a well-known narrative paradigm used by graphic novels. These dialogue lines may be used to convey additional information on how actors interact with artefacts and collaborate with other actors. Finally, textual attributes may be aggregated to scenes in the form of annotations and comments.

One particular characteristic of process stories is that, even though they can model traditional business processes with activities, flows and conditions, they can use scenes to convey other types of process-related information. For instance, scenes can be used to explain sensemaking and decision-making when performing activities. They can be used to add contextual details about the work setting, not only identifying the actors and artefacts involved but also other attributes and constraints like exchanged ideas or special requests. Scenes can also be used to express nuance, equivocality and conflict, reflecting past experiences, unusual scenarios, cautionary tales, which are typical of storytelling.

If we contrast this definition of process story with the traditional definitions of process models, e.g. the ones based on the dominant BPMN, we may easily notice the distinctions between their rich and lean imprints. Process stories are richer and open-ended while traditional models are leaner and formal. Very often, traditional process models describe the predominant flows but not the variations and exceptions, either because it is too expensive to model them, the language does not offer simple means to do it, or because models get too cluttered up to a point where they become useless. Modelling exceptions tends to be a difficult endeavour, since it may be difficult

to consider all different types of events and specific points in a model where they may occur. Furthermore, some complex aspects of work are difficult to model with traditional languages. Examples include flexibility (in assigning resources or shifting responsibilities), fuzzy connections between activities, performing continuous activities, jumping between activities, sharing information, and dealing with optional and ephemeral information [34]. Process stories avoid these problems by adopting a more open-ended approach.

We have also developed an innovative BPM modelling tool supporting the elicitation, analysis and design of process stories [31]. As noted above, the tool uses cartoons for eliciting and representing process stories. Users can select and configure cartoons from a database. The database provides a large collection of cartoons illustrating common business scenarios such as handing over a document, having a meeting, requesting/providing data and assigning a task. These cartoons can then be configured to express a specific business situation, e.g. indicating who participates in a meeting and what is decided there.

Furthermore, the tool supports a collaborative approach to process modelling. Even though individual scenes cannot be concurrently edited, teams can share process stories and modify the process-related information. This allows, for instance, expressing alternative flows, filling up gaps and enriching information with individual experiences. This tool can be contrasted with the more traditional business process modelling tools [35]. One striking difference is that while traditional tools mainly focus on activities, our tool concerns the open-ended environment surrounding them. Furthermore, while traditional tools emphasise formal conditions and flows, our tool emphasises informal sequences of events, which can be interpreted by readers using anthropomorphic information, thus affording implicitness and contextualisation.

Contemplating again the three research questions brought forward in the beginning of this section, we note the critical is the third one, what is the effectiveness of the tool and method. So far we have accomplished three rounds of experiments with the tool. In the following, we provide some insights from these experiments and present the obtained results.

3.1 First Round (Tool Usage)

The first round of experiments was primarily focussed on gathering formative insights about the tool usage. As previously discussed, the tool combines storytelling with cartoons, which breaks the traditional process modelling paradigm centred on activities, conditions and flows. The risk of users rejecting the tool because of a paradigm change was high and empirical tests were necessary to understand if the users would be able to develop process stories using the tool.

A set of individual modelling sessions were setup in a real-world organisation. The selected organization was seeking to integrate process management into an existing information system. However, they had not yet developed a clear process-oriented view and neither had started designing the process models. We approached the organization with two goals in mind, helping to select and design the processes and at the same time observing and analysing how some of its members would elaborate process stories

using the tool. The empirical tests were organized according to the following steps: meeting with leadership to identify and select processes; modelling sessions with key members using the tool; and analysing the outputs and obtaining informal feedback about the tool.

The modelling sessions were done in a period of four weeks. Different types of stakeholders were engaged in using the tool, including three managers. The form of engagement was different according to responsibilities. The three managers were engaged in individual modelling sessions, while the remaining 24 participants were divided in two groups and were assigned to joint sessions.

The results from these empirical tests were encouraging but also raised several major concerns about the tool and the modelling method [10]. One critical problem that was raised was the effort required for modelling process stories. One participant even referred to it as “mechanically slow”. The tool required picking scenes from the database, adding contextual information to each scene, and then organising scenes in a meaningful sequence. The participants complained the whole method required too many interactions and took excessive time. Further evidence suggested that this was a real constraint because most produced process stories were very short and lacked detail.

Two other concerns were also raised during the experiments. First, the participants revealed preoccupation with the correctness of their stories, i.e. how far they might diverge from the processes formalized by the organization. This suggested that organizational culture might also be a problem to consider when eliciting process stories.

Second, the participants were not always able to portray some situations as they wanted. Some of them tried to depict precise working contexts (e.g., a casual meeting taking place in a formal work area), while the tool offered a limited set of abstract scenes (e.g., casual meetings taking place in open spaces and formal meetings taking place in meeting rooms). As an exhaustive coverage of possible situations and contexts is hardly achievable, this suggested the participants should have been more exposed to storytelling strategies.

Overall, these tests indicated the concept of process story was appropriated by the participants but more training and repeated usage would be required to generate them; and also some positive reinforcements about the benefits of describing processes from alternative points of view would be necessary. Though the critical problem was the excessive effort required to tell a story. This led us to make structural changes in the tool to increase ease of use. The second and third rounds of experiments were done with the upgraded tool.

3.2 Second Round (Small Team, Desired Process)

The second round of experiments was targeted to a smaller organisation. It involved a small team of six persons, including the team leader. The team was responsible for providing a complex service related to information technology infrastructure management and the leader had arrived to the conclusion that service provision was affected by too many exceptions, ad hoc decisions and lack of knowledge management. In this particular case, the adoption of a process view was stimulated by the objectives of improving consistency, efficiency, transparency, accountability, and learning. In this

context, the leader decided to use the tool to design an improved business process model and the whole team was invited to participate.

The second round of experiments was designed in a more structured way. To start with, we defined a set of goals, questions and measurements, which is shown in Table 1. We considered three goals related to meaningfulness, contextualisation and sharing. Regarding meaningfulness, the intention was to assess if the generated process stories were sufficiently detailed and could be translated into purposeful activities. Asking if emotions, unexpected situations and contextual knowledge were present in process stories assessed contextualisation.

Concerning sharing, we looked for evidence of knowledge articulation and integration. This required dividing the process stories in different segments and analysing the respective levels of detail to find evidence of positive/negative changes.

The experiment was organised in three stages: training the participants on the tool usage; production of individual processes stories using the tool; and collaboration to reach a converged process story. The first phase lasted one week. The team received basic training on the tool usage and began using it for telling process stories. At this stage, there was frequent interaction between the team and the researchers to clarify the tool usage and to identify potential problems in developing process stories. This involved explaining the importance of scenes and how they could be configured to convey contextual information.

The second phase lasted about two weeks. The team members were invited to individually use the tool to elaborate their process stories. There was no interaction between the team and the researchers at this phase. Finally, in the final phase, participants were asked to collaboratively produce a converged process story. Since the tool allows viewing and changing each other's stories but does not support any explicit convergence process, the team would have to improvise a way for reaching a common, agreed upon story. This involved the team leader in gathering stories from all participants and suggesting a converged process to the team. The converged process would then be discussed and agreed by the team in a face-to-face meeting. Actually, because of the unanticipated complexity of some individual stories, two meetings were necessary to complete the discussion. After these two meetings, the team leader used the tool to record the collective process story.

The results from this experiment provided fine-grained information about our humanistic approach to process modelling [35]. Details about the individual process stories generated in phase two are shown in Tables 2, 3, 4, 5.

In Table 2, we summarise the measurements related to meaningfulness. Since the participants were purposely trying to model a desired process, not a current one, most stories scored poorly on the use of dialogue and highly on structural complexity. Two stories did not use dialogue at all, and all of them used structure as the primary means of telling a story. Most team members used narrative to describe what happened in a scene and for connecting scenes. Interestingly, every story could be converted into a traditional process model with activities, conditions and flows.

Table 3 summarises the obtained results regarding contextualisation. We note that few stories conveyed emotional elements such as uncertainty, frustration and disbelief. No story conveyed unexpected situations.

Table 1. Goals and measurements

Goal	Questions	Metric	Type	Data categories
Evaluate meaningfulness	Stories are detailed? Processes could be derived from stories?	Number of scenes	Quantitative	Numerical
		Use of narrative	Qualitative	Low, Medium, High
		Use of dialogue	Qualitative	Low, Medium, High
		Structural complexity	Qualitative	Low, Medium, High
		Conveys activities, conditions and flows	Qualitative	Yes, No
Evaluate contextualization	Stories portray emotion? Stories depict unexpected situations? Stories provide contextual knowledge?	Presence of emotional elements	Qualitative	Yes, No
		Presence of unexpected situations	Qualitative	Yes, No
		Presence of contextual reasoning	Qualitative	Low, Medium, High
Evaluate sharing	Stories helped the team better understand the process? Individual stories enriched the organisational practice?	Word count in story segments	Quantitative	Numerical
		Activity count in story segments	Quantitative	Numerical
		For each story segment, ratio of activities appearing in individual and converged stories	Quantitative	Numerical
		For all segments, ratio of activities appearing in individual and converged stories	Quantitative	Numerical

Tables 4 and 5 summarise how the process stories contributed to the final story through knowledge sharing. We note the participants tended to focus on particular

Table 2. Details about meaningfulness

Story #	Number of scenes	Use of dialogue	Use of narrative	Structural complexity	Story conveys activities, conditions and flows
1	10	None	Medium	Medium	Yes
2	8	Low	Medium	Medium	Yes
3	37	None	Medium	Very high	Yes
4	14	Medium	Medium	High	Yes
5	13	Medium	Low	High	Yes
6	15	Low	Medium	High	Yes

Table 3. Details about contextualisation

Story #	Presence of emotional elements	Presence of unexpected situations	Presence of contextual reasoning
1	No	No	Low
2	No	No	Low
3	No	No	Low
4	Yes	No	Low
5	Yes	No	Low
6	No	No	Low

Table 4. Details about sharing: word count (WC) and activity count (AC)

Story #	Segment 1		Segment 2		Segment 3		Segment 4	
	WC	AC	WC	AC	WC	AC	WC	AC
1	0	0	39	4	35	5	6	1
2	0	0	26	3	31	3	11	2
3	169	21	0	0	93	17	0	0
4	0	0	63	5	77	8	19	2
5	0	0	35	6	29	5	16	3
6	29	4	43	6	30	4	23	4

Table 5. Details about sharing: Ratios of activities appearing in individual and converged stories, shown by segment and overall. Stories not addressing a given segment are marked with “-”.

Story #	Segment 1	Segment 2	Segment 3	Segment 4	Overall
1	-	75 %	20 %	20 %	50 %
2	-	100 %	33 %	50 %	63 %
3	38 %	-	41 %	-	39 %
4	-	100 %	0 %	50 %	40 %
5	-	83 %	0 %	67 %	50 %
6	25 %	67 %	100 %	25 %	56 %

Table 6. Details about meaningfulness

# Story	Number of scenes	Use of dialogue	Use of narrative	Structural complexity	Story conveys activities, conditions and flows
1	2	55	2	Low	Yes
2	5	90	28	Medium	Yes
3	6	68	160	Medium	Yes
4	3	204	3	Medium	Yes
5	7	274	319	Medium	Yes
6	12	158	80	High	Yes
7	8	112	55	High	Yes
8	7	83	25	High	Yes
9	7	144	133	Medium	Yes
10	1	160	112	Medium	Yes
11	4	81	94	Low	Yes
12	7	118	74	Medium	Yes
13	3	105	78	Low	Yes
14	3	29	45	Low	Yes
15	2	7	8	Low	Yes
16	7	105	126	High	Yes
17	8	164	102	High	Yes
18	12	141	241	High	Yes
19	4	126	16	High	Yes
20	5	141	52	Medium	Yes

areas of expertise. For instance, story 3 concerned segments 1 and 3 but not 2 and 4; story 6 fully described story segment 3, but did not contribute much to the other segments. Perhaps more importantly, we also note that the converged story was assembled from diverse contributions of all stories in a rather balanced way: story 3 provided the lowest contribution but yet 39 % of the modelled activities were present in the converged story.

All in all, the second phase of experiments indicated the method and tool provided an effective approach for business process elicitation and modelling, but the generated process stories lacked contextualisation. On hindsight, the main explanation for the lack of contextualisation was related with the participants' goals. They were explicitly aiming at developing a new business process and therefore it is just natural that a new, idealised process does not convey much contextual information about a non-existing reality. In the third round of experiments we addressed that limitation.

3.3 Third Round (Large Team, Existing Process)

For the third round of experiments we selected a larger organisation. We have also chosen a complex business process involving multiple divisions; and involved more participants in telling process stories. The experimental design had to be adapted to accommodate the additional complexity. The goals and questions described in Table 1

were reused by this experiment; and a similar experimental design in three stages was followed. For the second stage, various modelling sessions were scheduled and the participants were invited to come up to one or more sessions for generating process stories. The participants would still work individually in these sessions. At the beginning of each session, the participants were informed about the process they should work on, but they were given freedom to model whatever they would consider relevant or interesting.

The third stage was also adapted, replacing the convergence meetings with a different approach, since converging a large, heterogeneous group is substantially more difficult than converging a small, homogeneous team. Instead, in the third stage we converted each individual process story into a traditional process model and then compared those stories with a reference process model previously approved by the organisations' management.

As reported in Table 6, we collected 20 stories in this experiment. We note the participants used narrative and dialogue as the primary means of telling their process stories, which is supported by the high word count regarding both narrative and dialogue. Most stories contained a relatively small number of scenes, which seems well aligned with the organisations' multi-divisional structure.

Some stories, even though having a low number of scenes, featured high structural complexity. As with the previous experiment, this suggests the participants externalised significant knowledge about the business process. Interestingly, stories 1, 11 and 13–15 provide low structural complexity but yet have significant use of dialogue and narrative. This suggests these scenes were used for storytelling. Even more interesting, story 10 is entirely contained in one scene with medium structural complexity, an indication of narrative sophistications.

Regarding contextualisation, we found a large number of stories depicting unexpected situations and emotional elements, which indicates the participants' interest in describing processes beyond the traditional activities, conditions and flows. We also observed a predominance of applied contextual reasoning in the vast majority of the collected stories (Table 7), including contextualised explanations supporting staff decisions over concrete circumstances, and detailed descriptions outlining unique scenarios that triggered custom behaviour/responses according to context. We argue that this combination, i.e. the depiction of unexpected situations together with emotional elements and contextually rich explanations (often foreign to the "happy path" normally depicted in mechanistic models), is an indicator of the externalization of participants' tacit knowledge in the form of process stories.

Table 8 provides a detailed summary of the process stories that were elaborated and their contributions to knowledge sharing. When comparing the activities described by the participants with the reference model (last line in Table 8), we can conclude that there is no direct mapping. Several stories provide significantly more knowledge, e.g. stories 3 and 4 more than double the number of activities. Again, this suggests that process stories enrich process knowledge with detailed insights about how work is actually done in the context of a business process.

In Table 8, we show two columns indicating if a story contributed to the reference model or not. As in the previous experiment, this provides another indication of how individual participants contributed to shared process knowledge. Once again, the

Table 7. Details about contextualisation

Story #	Depiction of unexpected situations	Presence of emotional elements	Presence of contextual reasoning
1	No	No	Low
2	Yes	No	High
3	No	No	Medium
4	Yes	Yes	High
5	No	No	High
6	Yes	No	High
7	Yes	No	High
8	Yes	No	High
9	Yes	Yes	High
10	Yes	Yes	High
11	No	Yes	High
12	No	No	Low
13	Yes	Yes	High
14	Yes	Yes	High
15	Yes	Yes	High
16	Yes	No	High
17	Yes	No	High
18	Yes	Yes	High
19	Yes	No	High
20	Yes	No	High

results support the view that process knowledge is a collective construction. The results also show if stories contradicted the reference process or not. We found out that five stories expressed knowledge contradicting the reference process sanctioned by the managers. This reinforces the idea that process stories can be richer than traditional business processes by expressing different and often contradicting views about a process.

4 Discussion

In this paper we suggest a classification of business processes in four categories: mechanistic, ad hoc, generative, and sensible. Sensible processes are founded on rich models and support predominant human control. We argue that such a combination leverages the capacity of humans and machines in BPM, which contributes to address the requirements of enriching knowledge [21] and flexibility in BPM [19, 20]. On the one hand, rich models afford information systems to reach beyond regular behaviour. For instance, rich models may provide details about process variations, exceptions, past occurrences, and contextual elements influencing the trajectory of a process instance. On the other hand, the predominance of human control in the interaction between humans and information systems affords more flexibility regarding process execution,

Table 8. Details about sharing (the last line provides details about the reference process model. Stories 4, 15, and 20 were omitted because they modelled a different process)

Story #	Number of activities per segment				Adds to reference	Contradicts reference
	Segment 2	Segment 2	Segment 3	Total		
1	0	0	7	7	yes	no
2	6	5	0	11	yes	no
3	7	8	0	15	yes	no
5	5	9	0	14	yes	no
6	0	0	27	27	yes	no
7	0	0	25	25	yes	yes
8	0	0	17	17	yes	no
9	13	0	0	13	yes	no
10	11	0	0	11	yes	yes
11	0	7	0	7	yes	no
12	2	0	15	17	yes	yes
13	5	0	0	5	yes	no
14	0	6	0	6	yes	no
16	14	0	0	14	yes	yes
17	13	2	9	24	yes	yes
18	25	2	0	27	yes	yes
19	13	0	0	13	yes	no
Reference	13	3	22	38	-	-

which may be supported with richer process models. Of course these possibilities depend on the capacity to design rich process models.

Having previously developed a process modelling tool supporting the design of rich process models, which we designate by process stories, in this paper we focus on a set of experiments that were set up to assess the capacity to design process stories and their potential value to organisations.

The several rounds of experiments demonstrated the validity of a set of assumptions behind process stories and the concept of sensible process. An important one is that process stories can be designed by end-users, i.e. business people that do not have expertise in process modelling. We argue that bringing process modelling to end-users increases process contextualisation. Traditional modelling tools usually require grasping specialised languages such as BPMN and UML. However, these languages tend to be formal, very complex and impose significant constraints, which are mainly related to their mechanistic lineage. The end result is that traditional modelling tools tend to be primarily used by modelling experts. Naturally, modelling experts have their own biases and goals when modelling business processes, which may conflict with the goals of the target organisations. The related literature refers to this phenomenon as silo views [36] and social distance [37]. Furthermore, existing process modelling languages and tools make it difficult to represent business rules [38, 39], collaborative aspects of business [40], and non-routine work [41]. The concept and information model

underlying process stories addresses these concerns by adopting a process modelling language that is informal, open-ended and closer to the business context. The results from the experiments support this argument, showing that end-users were able to develop process stories and the stories were relevant to discuss and elaborate process models.

Another important assumption behind process stories that was validated by the experiments is that they can bring about rich, contextualised information about the environment where they are enacted. Several stories developed in the experiments contained emotional elements, unexpected situations and contextual reasons. Furthermore, several stories also contained contradictory information, when compared with the reference process sanctioned by the management. However, we also noted that contextualisation may depend on the organisational goals. In the second experiment, where the participants were seeking to develop a desired process, the produced process stories did not contain contextual details. However, in the third experiment, where the participants were engaged in describing an existing process, the generated stories contained significant contextual information.

Finally, another relevant question about process stories is if they contribute or not to generate better process models. This question addresses matters of quality in general and effectiveness in particular. Ascertaining the quality of a business process is a complex endeavour, as it involves a large set of criteria like understandability, utility, efficiency, completeness, and correctness [42, 43], to name a few. In the specific context of modelling sensible processes, we argue that quality assessment should primarily concern matters related to model richness and human control. This suggests that aspects such as understandability and utility should prevail over more technical characteristics such as correctness and efficiency. Regarding the results from our experiments from this point of view, we note that in the second experiment, process stories helped teams agreeing on a process model that was more balanced than the individual stories. In the third experiment, the produced process stories simultaneously added to and contradicted the reference story, which suggests that process stories contributed to both comprehensibility and utility.

The concept of sensible business process opens up interesting avenues for future research. One interesting possibility is the transformation of process stories in traditional process models and subsequent integration in enterprise systems. In particular, process stories provide contextual information that may be relevant during process execution, for instance when handling exceptions. Another possibility, which is related to generative design, is the automated generation of a large number of alternative process models from a single process story, so that process participants and eventually enterprise systems could select a particular model depending on the specific conditions at hand. This would certainly contribute to increase the flexibility of enterprise systems.

References

1. van der Aalst, W.: Business process management: a comprehensive survey. *ISRN Softw. Eng.* **2013**, 1–37 (2013)

2. Ko, R., Lee, S., Lee, E.: Business process management (BPM) standards: a survey. *Bus. Process Manag. J.* **15**(5), 744–791 (2009)
3. Antunes, P., Mourão, H.: Resilient business process management: framework and services. *Expert Syst. Appl.* **38**(2), 1241–1254 (2011)
4. Mourão, H., Antunes, P.: Supporting effective unexpected exceptions handling in workflow management systems. In: Proceedings of the 22nd Annual ACM Symposium on Applied Computing, Special Track on Organizational Engineering, Seoul, Korea. ACM Press, pp. 1242–1249 (2007)
5. Sadiq, S., Orłowska, M., Sadiq, W.: Specification and validation of process constraints for flexible workflows. *Inf. Syst.* **30**(5), 349–378 (2005)
6. Hollnagel, E., Woods, D.: *Joint Cognitive Systems: Introduction to Cognitive Systems Engineering*. CRC Press, Boca Raton (2005)
7. Antunes, P., Zurita, G., Baloian, N., Sapateiro, C.: Integrating decision-making support in geocollaboration tools. *Group Decis. Negot.* **23**(2), 211–233 (2014)
8. Klein, G.: Naturalistic decision making. *Hum. Factors* **50**(3), 456–460 (2008)
9. Schmidt, R., Nurcan, S.: BPM and social software. In: Ardagna, D., Mecella, M., Yang, J. (eds.) *Business Process Management Workshops*. LNBP, vol. 17, pp. 649–658. Springer, Heidelberg (2009)
10. Antunes, P., Simões, D., Carriço, L., Pino, J.: An end-user approach to business process modeling. *J. Netw. Comput. Appl.* **36**(6), 1466–1479 (2013)
11. Rosemann, M.: Potential pitfalls of process modeling: part B. *Bus. Process Manag. J.* **12**(3), 377–384 (2006)
12. Chinosi, M., Trombetta, A.: Bpmn: an introduction to the standard. *Comput. Stand. Interfaces* **34**, 124–134 (2012)
13. Russell, N., van der Aalst, W., Hofstede, A., Wohed, P.: On the suitability of Uml 2.0 activity diagrams for Bp modelling. In: Proceedings of the 3rd Asia-Pacific Conference on Conceptual Modelling. Australian Computer Society, Inc., pp. 95–104 (2006)
14. Mayer, R.: *IDEF0 Function Modeling*. Air Force Systems Command (1992)
15. Andrews, T., Curbera, F., Dholakia, H., Golland, Y., Klein, J., Leymann, F., Liu, K., Roller, D., Smith, D., Thatte, S., Trickovic, I., Weerawarana, S.: *Business process execution language for web services version 1.1*. BEA, IBM, Microsoft, SAP, Siebel (2003)
16. Shapiro, R.: Xpdl 2.0: integrating process interchange and Bpmn. In: Fischer, L. (ed.) *Workflow Handbook*, pp. 183–194. Future Strategies Inc, Lighthouse Point (2006)
17. Momotko, M., Subieta, K.: Process query language: a way to make workflow processes more flexible. In: Benzúr, A.A., Demetrovics, J., Gottlob, G. (eds.) *ADBIS 2004*. LNCS, vol. 3255, pp. 306–321. Springer, Heidelberg (2004)
18. Schulte, S., Janiesch, C., Venugopal, S., Weber, I., Hoenisch, P.: Elastic business process management: state of the art and open challenges for BPM in the cloud. *Future Gener. Comput. Syst.* **46**, 36–50 (2015)
19. Reichert, M., Weber, B.: *Enabling Flexibility in Process-Aware Information Systems: Challenges, Methods, Technologies*. Springer, Heidelberg (2012)
20. Cabitza, F., Simone, C.: Computational coordination mechanisms: a tale of a struggle for flexibility. *Comput. Support. Coop. Work* **22**(4–6), 475–529 (2013)
21. Silva, A., Rosemann, M.: Processpedia: an ecological environment for BPM stakeholders' collaboration. *Bus. Process Manag. J.* **18**(1), 20–42 (2012)
22. Kabicher-Fuchs, S., Rinderle-Ma, S., Recker, J., Indulska, M., Charoy, F., Christiaanse, R., Mendling, J.: *Human-centric process-aware information systems (Hc-Pais)* (2012)
23. Reason, J.: *Managing the Risks of Organizational Accidents*. Ashgate, England (1997)
24. Antunes, P.: BPM and exception handling: focus on organizational resilience. *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.* **41**(3), 383–392 (2011)

25. McCarty, W.: *Modeling: a study in words and meanings. A Companion to Digital Humanities*, pp. 254–270. Blackwell, Malden (2004)
26. Weber, B., Reichert, M., Rinderle, S.: Change patterns and change support features – enhancing flexibility in process-aware information systems. *Data Knowl. Eng.* **66**(3), 438–466 (2008)
27. McCormack, J., Dorin, A., Innocent, T.: *Generative design: a paradigm for design research*. In: *Proceedings of Futureground, Design Research Society, Melbourne* (2004)
28. Antunes, P., Ferreira, A.: *Developing collaboration awareness support from a cognitive perspective*. In: *Proceedings of the 44th Hawaii International Conference on System Sciences (HICSS), Hawaii, 2011*. IEEE Computer Society
29. Antunes, P., Herskovic, V., Ochoa, S., Pino, J.: *Reviewing the quality of awareness support in collaborative applications*. *J. Syst. Softw.* **89**, 146–169 (2014)
30. Pries-Heje, J., Baskerville, R.: *The design theory Nexus*. *MIS Q.* **38**, 731–755 (2008)
31. Simões, D., Antunes, P., Pino, J.: *Humanistic approach to the representation of business processes*. In: *16th IEEE International Conference on Computer Supported Cooperative Work in Design (CSCWD), Wuhan, China*. IEEE, pp. 655–665 (2012)
32. Santoro, F., Borges, M., Pino, J.: *Acquiring knowledge on business processes from stakeholders’ stories*. *Adv. Eng. Inform.* **24**, 138–148 (2010)
33. Denning, S.: *The springboard: how storytelling ignites action in knowledge-era organizations*. *J. Organ. Change Manag.* **14**(6), 609–614 (2001)
34. Antunes, P., Herskovic, V., Ochoa, S., Pino, J.: *Modeling highly collaborative processes*. In: Shen, W., Li, W., Barthès, J. et al. (eds.) *17th IEEE International Conference on Computer Supported Cooperative Work in Design (CSCWD), Whistler, BC, Canada*. IEEE, pp. 184–189 (2013)
35. Simões, D., Antunes, P., Cranefield, J.: *Enriching knowledge in business process modelling: a storytelling approach*. In: Razmerita, L., Phillips-Wren, G., Jain, L. (eds.) *Innovations in Knowledge Management: The Impact of Social Media, Semantic Web and Cloud Computing*. Springer, Germany (2016)
36. Trkman, P.: *The critical success factors of business process management*. *Int. J. Inf. Manag.* **30**(2), 125–134 (2010)
37. Kolb, J., Zimoch, M., Weber, B., Reichert, M.: *How social distance of process designers affects the process of process modeling: insights from a controlled experiment*. In: *29th Symposium on Applied Computing (SAC 2014), Enterprise Engineering Track, Gyeongju, South Korea (24–28 March 2014)*
38. Kovacic, A.: *Business renovation: business rules (still) the missing link*. *Bus. Process Manag. J.* **10**(2), 158–170 (2004)
39. Green, P., Rosemann, M.: *Ontological analysis of integrated process models: testing hypotheses*. *Aust. J. Inf. Syst.* **9**(1), 30–38 (2001)
40. Caetano, A., Silva, A., Tribolet, J.: *Using roles and business objects to model and understand business processes*. In: *Proceedings of the 2005 ACM symposium on Applied computing, Santa Fe, New Mexico*. ACM, pp. 1308–1313 (2005)
41. Riemer, K., Johnston, R., Indulska, M.: *Questioning the philosophical foundations of business process modelling*. In: Gregor, S. (ed.) *Information Systems Foundations: Theorising in a Dynamic Discipline*, pp. 1–16. ANU E Press, Canberra (2014)
42. Recker, J.: *A socio-pragmatic constructionist framework for understanding quality in process modelling*. *Aust. J. Inf. Syst.* **14**(2), 43–62 (2007)
43. Mendling, J., Reijers, H., Recker, J.: *Activity labeling in process modeling: empirical insights and recommendations*. *Inf. Syst.* **35**(4), 467–482 (2010)

Contractual Proximity of Business Services

Lam-Son Lê^(✉)

Faculty of Computer Science and Engineering,
HCMC University of Technology, Ho Chi Minh City, Vietnam
lam-son.le@alumni.epfl.ch

Abstract. Business services arguably play a central role in service-based information systems as they would fill in the gap between the technicality of Service-Oriented Architecture and the business aspects captured in Enterprise Architecture. Business services have distinctive features that are not typically observed in Web services, e.g. significant portions of the functionality of business services might be executed in a human-mediated fashion. The representation of business services requires that we view human activity and human-mediated functionality through the lens of computing and systems engineering.

Given the specification of a relatively complex business service, practitioners can deal with its complexity either by breaking it down into constituent services through common practices such as outsourcing or delegation, or by picking up an existing group of services (e.g. from a service catalog) that best realize that functionality. To address these challenges, we devise a formal machinery to (a) verify if a group of services contractually match the specification of the larger service in question; (b) to assess the contractual proximity of service groups relative to a contractual service specification to help decide which combination of services from a catalog best realize the desired functionality.

Keywords: Service engineering · Goal modeling · Service contract · Quality of service · Service composition · Serviceability · Outsourcing

1 Introduction

In the last few years, service-oriented computing has become an emerging research topic in response to the shift from product-oriented economy to service-oriented economy. On the one hand, we now live in a growing services-based economy in which every product today has virtually a service component to it [1]. In this context, services are increasingly provided in different ways in order to meet growing customer demands. Business domains involving large and complex collection of loosely coupled services provided by autonomous enterprises are becoming increasingly prevalent [2, 3]. On the other hand, Information Technology (IT) has now been thoroughly integrated into our daily life [4] and gradually gives rise to the paradigm of ubiquitous computing. As such, business services

are essentially IT-enabled making the border between business services¹ and IT-enabled services blurred. At the high-level operationalization of a business service, we see business activities happening between service stakeholders. We may or may not witness IT operations at this representational level. At lower levels, the operationalization of these services are eventually translated into IT operations as we have seen in the cases of banking services, recruitment services, library services, auctioning services, etc.

From an IT perspective, there is a proliferation of methods and languages for Web services. Unfortunately, there has not been much work in modeling high-level services from a business perspective. The operationalization of business services has distinctive features that are not typically observed in plain Web services. Most notably, business services occur for a noticeable period of time, not spontaneously as Web services do. Their occurrences feature incremental human-mediated developments. As such, the representation of business services requires that we view human activity and human-mediated functionality through the lens of computing and systems engineering. Business services are typically operationalized by means of outsourcing or subcontracting, through which the provider of a relatively complex business service breaks it down into constituent services and subcontracts some of them to other service providers. Our study of real-life business services and collaboration with our industry partners reveal that, alternatively, the provider may pick up an existing set of services (e.g. from a service catalog) that best match it. The challenges here are to (i) verify if a group of services contractually match the specification of the big service in question; (ii) determine the preference in choosing a set of services from a service catalog in order to operationalize the big service in question.

In this paper, we devise a formal machinery to enable that verification and assessment as a continuation of our previous work on the representation of business services [5,6]. This work sheds light on the following research problems.

- (a) Given a certain set of service models, is a contract *serviceable*? We define *serviceability* as follows: a contract is serviceable using a set of services if and only the set of services represent a valid decomposition of the contract in question (which itself is viewed as a service)
- (b) Given a contract, what is the optimal service-level resourcing of the contract? This requires decomposing the given contract, which may be informally defined as generating from the contract in question a set of sub-contracts that cumulatively entail the obligations, commitments and schedules set forth by the contract. This contract decomposition should be guided by the available set of services.

Paper Structure. To make the paper self-contained, Sect. 2 briefly presents part of our previous work - a language for the contractual specification of business services. Sections 3 and 4 focus on the contribution of this work. Section 5

¹ By calling them business services, we mean services happening between people or business entities. They are enabled by IT in one way or another. For the sake of simplicity, we shall use the term “business service” or simply “service” to refer to these IT-enabled business services throughout this paper.

surveys related work and ends the paper by drawing some concluding remarks and outlining our future work.

2 Business Services Representation

Representing business services requires that we view human activity and human-mediated functionality through the lens of computing and systems engineering (and building a framework that is general enough to include both notions of services within its ambit). We are in need of an enhanced set of constructs that go beyond those that have been used for Web services. In the course of our research, we have found a close correlation between the notions of services and contracts (although the two notions are by no means identical). Our study of real-life business service descriptions, in domains as diverse as government services, IT services and consulting services, suggests that some contractual concerns appear routinely in the description of business services, and are part of the discourse on service design and re-design. In our view, business services should be contractually represented by taking into consideration the perspectives of both the service provider and service consumer(s) [6].

Table 1 summarizes service descriptors of a contract-oriented language² we specifically defined for business services. Figure 1 conceptualizes this language by means of meta-modeling.

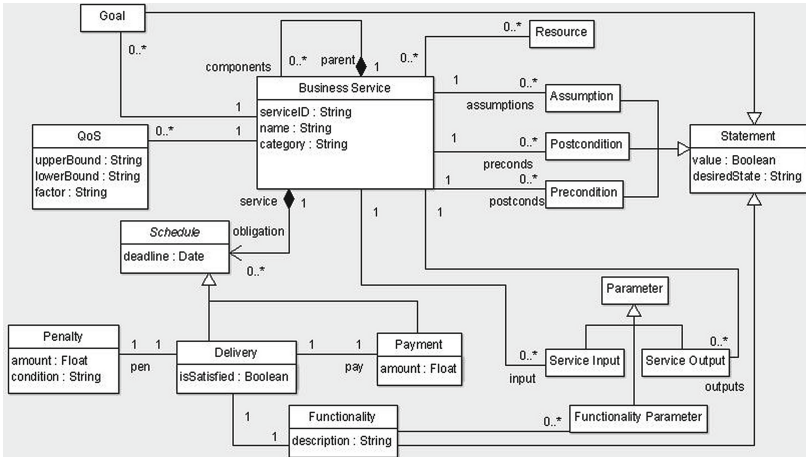


Fig. 1. Meta-model of a contract-oriented language for business services

² We have a full report of this work in a separate publication [5].

Table 1. Service descriptors of a contract-oriented language specifically defined for the representation of business services.

Descriptor	Definition
Goal	Intended effects or achievements of the service being represented
Precondition	Conditions that must hold to enable the occurrence of the service being represented
Postcondition	Effects or achievements of a service. They must hold upon the completion of the service being represented.
Assumption	Conditions on whose validity the occurrence of the service is contingent, but whose validity might not be verifiable when the service is invoked or during its execution
Input	Tangible or perceivable items that are usually fed by service consumers during the occurrence of the service being represented
Output	Tangible or perceivable items that are created or exchanged during the occurrence of the service being represented
QoS factor	Non-functional properties of the service being represented. Each Quality-of-Service (QoS) factor can be described in terms of the upper and lower bounds with quantitative evaluations.
Delivery	Incremental functionality during the occurrence of the service being represented. It may be described as a schedule in the form $\langle \textit{functionality}, \textit{deadline} \rangle$.
Payment	Incremental payment during the occurrence of the service being represented. It may be described as a schedule in the form $\langle \textit{amount}, \textit{deadline} \rangle$.
Penalty	A penalty is given when a functionality is not delivered as scheduled. It is usually associated with a delivery schedule.

3 Service Decomposition

We devise a formal machinery to verify if a service decomposition is valid as a solution to problem (a) mentioned in the introduction. Intuitively, the combination of constituent services can substitute for the decomposed one without affecting the existing business that the service consumers engage in. In other words, the constituent services altogether deliver at least the same whilst they expect no more than what the decomposed service does. We elaborate this proposition with respect to the service descriptors of our language (see Table 1) using the following running example.

Running Example. Let us consider a (passenger) car rental as a service denoted as S_1 . The provider of this service may break it down into three constituent services: (s_{11}) identity check & deposit; (s_{12}) vehicle pickup & return; (s_{13}) vehicle maintenance. The question is, given the contractual specifications of these services, is the decomposition $S_1 = \{s_{11}, s_{12}, s_{13}\}$ valid? In other words, is the contractual specification of S_1 serviceable by s_{11} , s_{12} and s_{13} ?

Table 2. Informal and formal representation of service goals. Note that s_{11} , s_{12} and s_{13} refer to services `identity check & deposit`, `vehicle pickup & return` and `vehicle maintenance`, respectively.

car rental's goals		constituent services' goals		
to provide customers with secured procedures for picking up and returning rental cars	$\forall r \in Customer, \forall c \in Car : booked(r, c) \rightarrow creditcardDeposited(c)$	$\forall r \in Customer, \forall v \in Vehicle : booked(r, v) \rightarrow (creditcardDeposited(c) \vee bankDeposited(c))$	to provide customers with convenient yet secured procedures for picking up and returning rental vehicles	s_{11}
to ensure customers who booked in advance will be able to pick up cars they selected	$\forall r \in Customer, c \in Car : booked(r, c) \rightarrow proceedWithPickup(r, c)$	$\forall r \in Customer, v \in Vehicle : booked(r, v) \rightarrow proceedWithPickup(r, v)$	to ensure customers who booked in advance will be able to pick up vehicles they selected	s_{12}
to make sure that all cars are mechanically sound before handing them over tenants	$\forall r \in Customer, c \in Car : booked(r, c) \rightarrow mecSound(c)$	$\forall r \in Customer, v \in Vehicle : booked(r, v) \rightarrow mecSound(v)$	to make sure that all vehicles are mechanically sound before handing them over tenants	s_{12}, s_{13}
to computerize the pickup and return procedures and to digitize rental records	$\forall c \in Car : rentalDigitized(c)$	$\forall v \in Vehicle : rentalDigitized(v) \wedge historyDigitized(v)$	to computerize the pickup and return procedures and to digitize rental records as well as service history of all vehicles	s_{11}, s_{13}

3.1 Goal

Following [7], assuming that a set of goals refers to the conjunction of its elements, we define goal refinement in Definition 1.

Definition 1. Goal G is refined into a set of sub-goals $\{g_1, g_2, \dots, g_n\}$ to be valid if and only if:

- $g_1 \wedge g_2 \wedge \dots \wedge g_n \not\models \perp$
- $g_1 \wedge g_2 \wedge \dots \wedge g_n \models G$
- $G' \not\models G$ for any $G' \subset \{g_1, g_2, \dots, g_n\}$

Example 1. Table 2 lists goals of service `car rental` in its leftmost two columns. Corresponding goals of services `identity check & deposit`, `vehicle pickup & return` and `vehicle maintenance` are given in the rightmost two columns. Note that all goals are described both informally (in natural language) and formally (by means of first-order logic). Given $Car \subset Vehicle$ and a few deduction rules in first-order logic [8], we can straightforwardly prove that the constituent services' goals actually refine `car rental`'s goals according to Definition 1.

3.2 Precondition/Postcondition

The constituent services altogether require same or weaker preconditions while producing same or stronger postconditions as the decomposed service does. This proposition could formally be interpreted as follows.

- The pre-condition of the decomposed service entails those of its constituent services, formally $pre_S \models pre_1 \wedge pre_2 \wedge \dots \wedge pre_n$ where $pre_1, pre_2, \dots, pre_n$ denote preconditions of constituent services; pre_S denotes the precondition of the decomposed service.
- The post-conditions of constituent services entail that of the decomposed service, formally $post_1 \wedge post_2 \wedge \dots \wedge post_n \models post_S$ where $post_1, post_2, \dots, post_n$ denote postconditions of constituent services; $post_S$ denotes the postcondition of the decomposed service.

3.3 Assumption

The standpoint on service assumption is similar to that on service precondition. Intuitively, the constituent services altogether should make same or weaker as the decomposed service does. This proposition could be formalized as follows. The assumption of the decomposed service entails those of its constituent services, formally $asmp_S \models asmp_1 \wedge asmp_2 \wedge \dots \wedge asmp_n$ where $asmp_1, asmp_2, \dots, asmp_n$ denote assumptions made for constituent services; $asmp_S$ denotes the assumption made for the decomposed service.

Example 2. One of the assumptions of service **car rental** is about the customer’s responsibility: tenants must check (and top up if necessary) engine oil and other fluids of their rental car especially during long drives and will be held liable for any breakdown caused by the insufficiency of oil and/or fluid. If the constituent service **vehicle maintenance** ensures that all vehicles are equipped with appropriate level sensors, it can bear on the following weaker assumption without affecting the serviceability of the contractual spec of service **car rental**: tenants must top up engine oil and other fluids of their rental car whenever oil/fluid level warnings are given on their car dashboard.

3.4 Input/Output

The constituent services altogether require same or less input while produce same or more output as the decomposed service does. The intuitive meaning of “less input” can be explained as (i) constituent services take a subset of input objects taken by the decomposed service; or (ii) constituent services take the same number of input objects but some of their input objects subsume input objects of the decomposed services. Similarly, “more output” actually means (i) constituent services produce a superset of output objects produced by the decomposed service; or (ii) constituent services produce the same number of output objects but some of their output objects can be substituted for output objects of the decomposed services.

The notion of substitutability was made popular in object-oriented programming and later extended to the context of object-oriented conceptual modeling [9]. An object can substitute for another if the former can be safely used in a context where the latter is expected. This proposition is formulated in Definition 2. As an example, a passenger car can substitute for an vehicle. The type that describes all vehicles subsumes the type describing passenger cars.

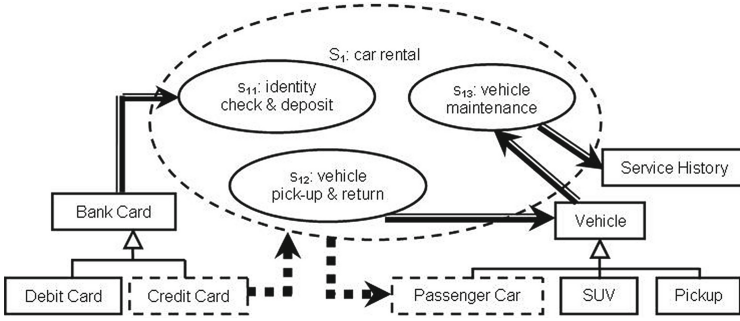


Fig. 2. The contractual spec of service *car rental* is serviceable by services *identity check & deposit*, *vehicle pickup & return* and *vehicle maintenance* in terms of input/output because the input and output objects of the constituent services subsume those specified for the decomposed service.

Definition 2. Object *s* can substitute for object *q*, denoted as $Type(s) <: Type(q)$, if object *s* belongs to a subtype of or the same type as what object *q* belongs to. The type of *q* is said to subsume the type of *s*.

Whether a set of constituent services match a decomposed service in terms of input/output can be formulated as follows. Example 3 illustrates this formulation.

- An input object taken by the constituent services is either passed by the decomposed service or substitutable for by a corresponding input object taken by the decomposed service. Formally, we have $\forall x : x \in \bigcup_{i=1}^n input_i \rightarrow Pass(S, x) \vee (\exists x' : x' \in input_S \wedge \neg Pass(S, x') \wedge Type(x') <: Type(x))$ where $Pass(serv, o)$ implies that service *serv* produces object *o* and passes it as an input object to another service; $input_S$ denotes the set of input objects of the decomposed service and $input_i$ the set of input objects of the i^{th} constituent service.
- For every output object produced by the decomposed service, there is a corresponding output object produced by one of the constituent services that it can substitute for. Formally, we have $\forall x : x \in output_S \rightarrow \exists x' : x' \in \bigcup_{i=1}^n output_i \wedge Type(x) <: Type(x')$ where $input_S$ denotes the set of input objects of the decomposed service and $input_i$ the set of input objects of the i^{th} constituent service. Note that some output objects produced by the constituent services may be consumed by the decomposed service.

Example 3. The constituent services s_{11} , s_{12} , s_{13} may accept input objects and produce the output objects of different types than the main service S_1 does. Specifically, s_{11} can accept not only credit card but also debit card (e.g. Maestro) whilst s_{12} , s_{13} can deal with not only passenger cars but also utility cars (e.g. 4 × 4, pickup). In addition, s_{13} produces more output objects (i.e. **Service History**) than S_1 does. This service decomposition is still valid in terms of input

and output despite the difference of input/output types between the constituent services and the main one.

Figure 2 depicts the input and output objects of these services. We use dashed lines to draw the main service `car rental` and its input/output objects. The constituent services and their input/output objects are drawn under solid lines. Double line arrows show input or output (depending on the direction of the arrows) of services. Triangle-headed arrows diagrammatically illustrate the subtyping relation between input/output objects.

3.5 QoS Factor

The QoS factors of the decomposed service must be satisfied by those of the constituent services. This proposition will be formally elaborated in light of the semiring-based representation of QoS factors. Definition 3 defines *semiring* [10] that will be used to express QoS factors.

Definition 3. *A semiring is a tuple $\langle A, \oplus, \otimes, 0, 1 \rangle$ such that*

- A is a set and $0, 1 \in A$
- \oplus , called the additive operation, is a commutative, associative operation having 0 as its unit element (i.e. $a \oplus 0 = a = 0 \oplus a$)
- \otimes , called the multiplicative operation, is an associative operation such that 1 is its unit element and 0 is its absorbing element (i.e. $a \otimes 0 = 0 = 0 \otimes a$)
- \otimes distributes over \oplus (i.e. $\forall a, b, c \in A \rightarrow a \otimes (b \oplus c) = a \otimes b \oplus a \otimes c$)

An idempotent semiring is a semiring whose additive operation is idempotent (i.e. $a \oplus a = a$). To make an idempotent semiring applicable for the representation of QoS, we endow it with a canonical order defined as $a \preceq b$ iff $a \oplus b = b$. A semiring is used to express the domain and the order between values that feature a QoS. To represent QoS factors, we may use the notion of *bounded lattice*. Each bounded lattice has a greatest element (denoted as \top) and a least element (denoted as \perp) and features two operations: meet (denoted as \wedge) and join (denoted as \vee) [11].

Example 4. Service S_1 (`car rental`) and its constituent service s_{12} (`vehicle pickup & return`) has a QoS factor in common, which is about the likelihood that the customers will be handed a car of the specific type they have booked. S_1 aims to satisfy their customers' bookings at the rate of 85% or more while service s_{12} features a booking satisfaction rate of at least 95%. The domain of these two QoS factors is represented using an idempotent semiring denoted as $\langle [0, 1], \max, \times, 0, 1 \rangle$. This semiring takes the max function as its additive operation and the classical multiplication as its multiplicative operation. Intuitively, the max function of this semiring can be used to express a canonical order and the \times operator is useful for reasoning over probability. Note that $[0, 1]$ denotes an interval of real numbers that represents the probability domain. This semiring is endowed with the classical comparison as its canonical order.

Table 3. An example of how QoS factors can be expressed using semirings and lattices

Service QoS	Semiring $\langle A, \oplus, \otimes, 0, 1 \rangle$	Canonical order	Lattice $\langle \perp, \top, \wedge, \vee \rangle$
s_{12} : satisfaction rate of 95 %	$\langle [0, 1], max, \times, 0, 1 \rangle$	Real numbers comparison	$\langle 0.95, 1, min, max \rangle$
S_1 : satisfaction rate of 85 %	$\langle [0, 1], max, \times, 0, 1 \rangle$	Real numbers comparison	$\langle 0.85, 1, min, max \rangle$

To this end, comparing the two QoS factors of these two services would boil down to checking if the lattice that represents the QoS factor of one service is a *sublattice* of the lattice that represents the corresponding QoS factor of the other service. As illustrated in Table 3, the QoS factor of s_{12} entails that of S_1 because the lattice created for s_{12} is in fact a sublattice of the lattice created for S_1 .

Example 4 motivates us to define whether a set of constituent services fulfill the decomposed service in terms of QoS factors as follows. A set of constituent services fulfill the decomposed service in terms of QoS factors iff for each QoS factor of the decomposed service S , the following two clauses hold

- there is a corresponding QoS factor that is specified for the constituent service s_i of which the domain is represented using the same semiring
- the lattice that characterizes the QoS factor of S is a sublattice of the corresponding lattice for s_i

3.6 Delivery

We come up with a notion of *functionality entailment*. At any given time, the combined functionality delivered by the constituent services must entail the functionality specified for the decomposed service. Formally, we have $\forall t \in Time : acc(\{f_1(t), f_2(t), \dots, f_n(t)\}) \models f_S(t)$ where $f_i(t)$ denotes the functionality delivered by the i^{th} constituent service at the moment t and $f_S(t)$ signifies the functionality tentatively delivered at the moment t by the decomposed service. The function *acc* yields cumulative effects of the functionalities of all constituent services. This accumulation of service functionalities can be reasoned in an analogous way to how tasks and activities are put together in a business process model [12].

3.7 Penalty

Penalties are given if a service functionality is not delivered as scheduled. Penalties can be applied to constituent services as well as the decomposed service. Basically, if a penalty is given because the functionality of the decomposed service is not delivered as scheduled, the provider of this service will do root-cause analysis to find out what service providers of constituent services should be given

a derived penalty. To avoid being overly charged, the provider of the decomposed service will give other service providers appropriate penalties. There are two basic approaches for monitoring the penalties. In the first approach, at every critical moment (that is usually called check point in the realm project management [13]), derived penalties given to service providers of constituent services must surpass the penalty given to the main service provider. Formally, we have $\forall t \in Time : \sum_{i=1}^n pmlt(f_i(t)) > pmlt(f_S(t))$ where $pmlt$ is a function that defines a penalty given based on the functionality delivered. We call this the *linear* approach.

In the second approach, the main service provider may accept a deficit in terms of penalties during the occurrence of the service being decomposed. However, the accumulated penalties given to providers of constituent services must surpass the penalty that is given to the decomposed service. Formally, we have $\sum_{i=1}^n acc(pmlty(f_i)) > acc(pmlt(f_S))$. We call this the *accumulative* approach.

3.8 Payment

The service provider will ultimately take profit in outsourcing constituent services. In other words, the total payments the provider makes for outsourced services should be less than the payment received from the service consumers of the main service. The two monitoring approaches discussed in Subsect. 3.7 also apply to payments. In the linear approach, we have $\forall t \in Time : \sum_{i=1}^n pay(f_i(t)) > pay(f_S(t))$ where pay denotes the payment for a given functionality delivered. In the cumulative approach, the main service provider may accept a deficit in terms of payments during the occurrence of the service being decomposed. This can be formulated as $\sum_{i=1}^n acc(pay(f_i)) > acc(pay(f_S))$.

4 Towards Operationalization Preference and Contractual Proximity of Business Services

In this section, we address problem (b) that is articulated in the introduction. We define partial orders among service groups that fully meet a given contractual specification (Subsect. 4.1) and between those that do not (Subsect. 4.2). These definitions will potentially lead to selection criteria to assist the service provider in deciding which service group to be selected among those that are available from a service catalog in order to operationalize the given contractual specification.

4.1 Operationalization Preference

Suppose that we have multiple choices in picking up an existing set of services to match a given contractual service specification. In order to decide how to operationalize this contractual specification, the potential service provider for this contractual specification needs to know which service group is the best. We investigate the preference among groups of services that all satisfy a contractual

Table 4. A set of services $SS_1 = \{s_{11}, s_{12}, \dots, s_{1n}\}$ is preferred over another set $SS_2 = \{s_{21}, s_{22}, \dots, s_{2m}\}$ that both satisfy a contractual service specification S .

Descriptor	Denotation	Definition
Goal	$SS_1 \succeq_{goal(S)} SS_2$	$goal_{11} \wedge goal_{12} \wedge \dots \wedge goal_{1n} \models goal_{21} \wedge goal_{22} \wedge \dots \wedge goal_{2m} \models goals$
Precondition	$SS_1 \succeq_{pre(S)} SS_2$	$pre_S \models pre_{21} \wedge pre_{22} \wedge \dots \wedge pre_{2m} \models pre_{11} \wedge pre_{12} \wedge \dots \wedge pre_{1n}$
Postcondition	$SS_1 \succeq_{post(S)} SS_2$	$post_{11} \wedge post_{12} \wedge \dots \wedge post_{1n} \models post_{21} \wedge post_{22} \wedge \dots \wedge post_{2m} \models posts$
Assumption	$SS_1 \succeq_{asmp(S)} SS_2$	$asmp_S \models asmp_{21} \wedge asmp_{22} \wedge \dots \wedge asmp_{2m} \models asmp_{11} \wedge asmp_{12} \wedge \dots \wedge asmp_{1n}$
Input	$SS_1 \succeq_{input(S)} SS_2$	$\forall x : x \in \bigcup_{i=1}^n input_{1i} \rightarrow Pass(S, x) \vee (\exists x' : x' \in \bigcup_{j=1}^m input_{2j} \wedge \exists x'' : x'' \in input_S \wedge \neg Pass(S, x') \wedge Type(x'') < Type(x') < Type(x))$
Output	$SS_1 \succeq_{output(S)} SS_2$	$\forall x : x \in \bigcup_{j=1}^m output_{2j} \rightarrow \exists x' : x' \in \bigcup_{i=1}^n output_{1i} \wedge \exists x'' \in output_S \wedge Type(x) < Type(x') < Type(x'')$
QoS	$SS_1 \succeq_{qos(S)} SS_2$	for each QoS constraint of a member service $s_{1i} \in SS_1$, the following two clauses hold <ul style="list-style-type: none"> – there is a corresponding QoS constraint that is specified for a member service of $s_{2j} \in SS_2$ and another corresponding QoS constraint specified for S of which the domains are represented using the same semiring; – the lattice that characterizes the QoS constraint of s_{1i} is a sublattice of the corresponding lattice for s_{2j}, which is another sublattice of the corresponding lattice for S
Delivery	$SS_1 \succeq_{delivery(S)} SS_2$	$\forall t \in Time : acc(\{f_{11}(t), f_{12}(t), \dots, f_{1n}(t)\}) \models acc(\{f_{21}(t), f_{22}(t), \dots, f_{2m}(t)\}) \models f_S(t)$
Payment	$SS_1 \succeq_{payment(S)} SS_2$	$\forall t \in Time : \sum_{i=1}^n pay(f_{1i}(t)) > \sum_{j=1}^m pay(f_{2j}(t)) > pay(f_S(t))$
Penalty	$SS_1 \succeq_{penalty(S)} SS_2$	$\forall t \in Time : \sum_{i=1}^n pnlt(f_{1i}(t)) > \sum_{j=1}^m pnlt(f_{2j}(t)) > pnlt(f_S(t))$

specification. The preference is defined with respect to an individual service descriptor.

Table 4 gives formal definitions for this preference. A set of service SS_1 is preferred over another set SS_2 can be interpreted as if SS_1 satisfies an imaginary contractual specification represented by SS_2 , in pretty much the same manner we define how SS_1 meets a contractual specification S in Sect. 3. Note that the preference between two sets of services with respect to input/output is defined based on Theorem 1 that is stated and proved as follows.

Theorem 1. *Substitutability as defined in Definition 2 is reflexive and transitive.*

Proof. For any object \mathbf{s} , we obviously have $Type(\mathbf{s}) < Type(\mathbf{s})$. So \mathbf{s} can substitute for itself. Suppose object \mathbf{s} can substitute for object \mathbf{p} , which can in turn substitute for object \mathbf{q} . According to Definition 2, we have $Type(\mathbf{s}) < Type(\mathbf{p})$ and $Type(\mathbf{p}) <$

$Type(q)$ meaning that any proposition that holds for object q 's type will hold for object p 's type and then for object s 's type too. So $Type(s) <: Type(q)$ meaning s can substitute for q .

Each preference relationship defined in Table 4 features a partial order between sets of services in the sense that, between two sets of services that both satisfy a contractual specification, we may or may not firmly determine that one is preferred over the other.

4.2 Contractual Proximity

Given two sets of services that do not satisfy the contractual service specification S , we need to measure how far away they are different from S to determine which one is preferred over the other. We call this measure the *contractual proximity*.

Table 5 formally defines the contractual proximity of a set of business services with respect to a certain contractual service specification for individual service descriptors. The rationale behind this formal definitions is to determine smallest sets (for input and output), weakest conditions (for goal, precondition, postcondition and assumption) or minimal values (for payment and penalty) that make a set of services satisfy the given contractual service specification.

We now proceed in defining the preference between two sets of business services that do not satisfy a contractual service specification. Table 6 gives formal definitions of whether a set of services $CS_1 = \{s_{11}, s_{12}, \dots, s_{1n}\}$ is preferred over another set of service $CS_2 = \{s_{21}, s_{22}, \dots, s_{2m}\}$ with respect to a contractual service specification S . This preference is defined based on the definitions of contractual proximity given in Table 5. Intuitively speaking, between two sets of services that do not meet a contractual specification, the one that is closer to this specification will be preferred over the other.

5 Related Work and Conclusion

In our group, we do research in the modeling, formal representation and the strategic alignment of IT-enabled *business* services. In our view, contractual concerns appear routinely in the description of business services, and are part of the discourse on service design and re-design. We have defined a representational language dedicatedly for business services in a contract-oriented perspective [5]. We are interested in the *serviceability* of contractual service specifications and motivated by the following two problems (i) to verify a set of services against a certain contractual specification; (ii) to determine the preference in choosing a set of services from a service catalog in order to operationalize the contractual specification. In this paper, we present a formal machinery to verify the decomposition of services and assess the *contractual proximity* of business services against a contractual specification. The originality of our work is that we take into account the incremental nature of business services (i.e. delivery schedules, payment schedules, penalties) as well as human-mediated factors (i.e. QoS, assumption, goals) of business services in addition to service descriptors that

Table 5. Definition of contractual proximity of a set of services $CS = \{s_1, s_2, \dots, s_n\}$ with respect to a contractual service specification S .

Descriptor	Denotation	Definition
Goal	$CS\Delta_{goal}S$	some goal gl s.t. $- gl \wedge goal_1 \wedge goal_2 \wedge \dots \wedge goal_n \models goal_S$ $- \forall gl' : (gl' \wedge goal_1 \wedge goal_2 \wedge \dots \wedge goal_n \models goal_S) \rightarrow (gl' \models gl)$
Precondition	$CS\Delta_{pre}S$	some condition $cond$ s.t. $- pre_S \wedge cond \models pre_1 \wedge pre_2 \wedge \dots \wedge pre_n$ $- \forall cond' : (pre_S \wedge cond' \models pre_1 \wedge pre_2 \wedge \dots \wedge pre_n) \rightarrow (cond' \models cond)$
Postcondition	$CS\Delta_{post}S$	some condition $cond$ s.t. $- cond \wedge post_1 \wedge post_2 \wedge \dots \wedge post_n \models post_S$ $- \forall cond' : (cond' \wedge post_1 \wedge post_2 \wedge \dots \wedge post_n \models post_S) \rightarrow (cond' \models cond)$
Assumption	$CS\Delta_{asmp}S$	some assumption Ap s.t. $- asmp_S \wedge Ap \models asmp_1 \wedge asmp_2 \wedge \dots \wedge asmp_n$ $- \forall Ap' : (asmp_S \wedge Ap' \models asmp_1 \wedge asmp_2 \wedge \dots \wedge asmp_n) \rightarrow (Ap' \models Ap)$
Input	$CS\Delta_{input}S$	a set of objects $In = \{x x \in \bigcup_{i=1}^n input_i \wedge \neg Pass(S, x) \wedge \neg \exists x' \in input_S : Type(x') < Type(x)\}$
Output	$CS\Delta_{output}S$	a set of objects $Out = \{x x \in output_S \wedge \neg \exists x' \in \bigcup_{i=1}^n output_i : Type(x) < Type(x')\}$
QoS	$CS\Delta_{qos}S$	a set of QoS constraints of any member service $s_i \in CS$ s.t. $-$ there is no corresponding QoS constraint that is specified for S of which the domains can be represented using the same semiring; or $-$ the lattice that characterizes the QoS constraint of s_i is not a sublattice of the corresponding lattice for S
Delivery	$CS\Delta_{delivery}S$	a function $dd(t)$ s.t. $- dd(t) \wedge acc(\{f_1(t), f_2(t), \dots, f_n(t)\}) \models f_S(t)$ $- \forall fct : (fct \wedge acc(\{f_1(t), f_2(t), \dots, f_n(t)\}) \models f_S(t)) \rightarrow (fct \models dd(t))$ where $t \in Time$
Payment	$CS\Delta_{payment}S$	a function $dpay(t) = \sum_{i=1}^n pay(f_i(t)) - pay(f_S(t))$ where $t \in Time$
Penalty	$CS\Delta_{penalty}S$	a function $dplnt(t) = \sum_{i=1}^n plnt(f_i(t)) - plnt(f_S(t))$ where $t \in Time$

have been commonly addressed in the field of service-oriented computing such as input/output, pre/post conditions.

To the best of our understanding, service contracts have been studied in the context of Web service evolution [14] and the integration of heterogeneous

Table 6. A set of services $CS_1 = \{s_{11}, s_{12}, \dots, s_{1n}\}$ is preferred over another $CS_2 = \{s_{21}, s_{22}, \dots, s_{2m}\}$ with respect to a service specification S for individual service descriptors.

Descriptor	Denotation	Definition
Goal	$CS_1 \succeq_{goal} CS_2$	$CS_2 \Delta_{goal} S \models CS_1 \Delta_{goal} S$
Precondition	$CS_1 \succeq_{pre} CS_2$	$CS_2 \Delta_{pre} S \models CS_1 \Delta_{pre} S$
Postcondition	$CS_1 \succeq_{post} CS_2$	$CS_2 \Delta_{post} S \models CS_1 \Delta_{post} S$
Assumption	$CS_1 \succeq_{asmp} CS_2$	$CS_2 \Delta_{asmp} S \models CS_1 \Delta_{asmp} S$
Input	$CS_1 \succeq_{input} CS_2$	$CS_1 \Delta_{input} S \subseteq CS_2 \Delta_{input} S$
Output	$CS_1 \succeq_{output} CS_2$	$CS_1 \Delta_{output} S \subseteq CS_2 \Delta_{output} S$
QoS	$CS_1 \succeq_{qos} CS_2$	$CS_1 \Delta_{qos} S \subseteq CS_2 \Delta_{qos} S$
Delivery	$CS_1 \succeq_{delivery} CS_2$	$CS_2 \Delta_{delivery} S \models CS_1 \Delta_{delivery} S$
Payment	$CS_1 \succeq_{payment} CS_2$	$CS_1 \Delta_{payment} S \leq CS_2 \Delta_{payment} S$
Penalty	$CS_1 \succeq_{penalty} CS_2$	$CS_1 \Delta_{penalty} S \leq CS_2 \Delta_{penalty} S$

services [15]. Our work differs from theirs primarily in ways IT-enabled business services are contractually represented. We stress that human-mediated services should contractually be specified as opposed to being programmatically described like Web services. Our research also differs from work on integrating human activities to business processes like BPEL4People. We take a rather declarative approach while BPEL4People is essentially imperative. In terms of QoS representation, we note that semiring has been exploited in expressing QoS factors [16, 17]. In our approach, we propose to determine a lattice of semiring that best represents a given QoS factor, in order to make the comparison of QoS factors technically possible.

Work is currently underway to map our formal definitions to Alloy - a lightweight formal, declarative modeling language [18]. The goal is to develop a computer-interpretable engine that computerize our formal machinery. This will open the door for the implementation of a toolkit that manages a service repository and permits reasoning on service contracts (e.g. verifying a set of services against a contract, ranking alternative sets of services based on their serviceability on a specific contract).

Acknowledgment. I would like to thank my former colleague, Aditya Ghose, for his valuable feedback on this work, especially in the formalization of service goals and QoS. I am also thankful to the other members of his research group for their comments (mostly about terminology used in modeling business services) on my talks given in the weekly seminars when I was in this group.

References

1. Paulson, L.D.: Services science: a new field for today's economy. *Computer* **39**(8), 18–21 (2006)
2. Singh, M.P., Huhns, M.N.: *Service-Oriented Computing: Semantics, Processes*. Wiley, Agents (2005)
3. IfM and IBM: *Succeeding through Service Innovation: A Service Perspective for Education, Research*. University of Cambridge Institute for Manufacturing, Cambridge, UK, Business and Government. White paper (2008)
4. Hansmann, U., Merk, L., Nicklous, M.S., Stober, T.: *Pervasive Computing: The Mobile World*, 2nd edn. Springer, Heidelberg (2011)
5. Ghose, A.K., Lê, L.S., Hoesch-Klohe, K., Morrison, E.: The business service representation language: a preliminary report. In: Cezon, M., Wolfsthal, Y. (eds.) *ServiceWave 2010 Workshops*. LNCS, vol. 6569, pp. 145–152. Springer, Heidelberg (2011)
6. Lê, L.S., Dam, H., Ghose, A.: On Business services representation - the 3 x 3 x 3 approach. In: *Proceedings of 21st Australasian Conference on Information Systems*, Brisbane, Australia, Association for Information Systems, December 2010
7. Letier, E., van Lamsweerde, A.: Deriving operational software specifications from system goals. *ACM SIGSOFT Softw. Eng. Notes* **27**(6), 119–128 (2002)
8. Smullyan, R.M.: *First-Order Logic*. Dover Publications, Mineola (1995)
9. Liskov, B.H., Wing, J.M.: A behavioral notion of subtyping. *ACM Trans. Program. Lang. Syst.* **16**(6), 1811–1841 (1994)
10. Hardouin, L., Cottenecau, B., Lhommeau, M., Le Corronc, E.: Interval systems over idempotent semiring. *Linear Algebra Appl.* **431**(5–7), 855–862 (2009)
11. Davey, B.A., Priestley, H.A.: *Introduction to Lattices and Order*. Cambridge University Press, Cambridge (2002)
12. Hinge, K., Ghose, A., Koliadis, G.: Process SEER: a tool for semantic effect annotation of business process models. In: *Proceedings of the 13th IEEE International Conference on Enterprise Distributed Object Computing*, Auckland, New Zealand, pp. 49–58. IEEE Computer Society, September 2009
13. Hughes, B., Cotterell, M.: *Software Project Management*, 5th edn. McGraw-Hill, New Delhi (2009)
14. Andrikopoulos, V., Benbernou, S., Papazoglou, M.P.: Evolving services from a contractual perspective. In: van Eck, P., Gordijn, J., Wieringa, R. (eds.) *CAiSE 2009*. LNCS, vol. 5565, pp. 290–304. Springer, Heidelberg (2009)
15. Comerio, M., Truong, H.-L., De Paoli, F., Dustdar, S.: Evaluating contract compatibility for service composition in the SeCO₂ framework. In: Baresi, L., Chi, C.-H., Suzuki, J. (eds.) *ICSOC-ServiceWave 2009*. LNCS, vol. 5900, pp. 221–236. Springer, Heidelberg (2009)
16. Hirsch, D., Tuosto, E.: SHReQ: coordinating application level QoS. In: *3rd IEEE International Conference on Software Engineering and Formal Methods*, Koblenz, Germany, pp. 425–434, September 2005
17. Ferrari, G., Lluch-Lafuente, A.: A logic for graphs with QoS. *Electron. Notes Theor. Comput. Sci.* **142**, 143–160 (2006)
18. Jackson, D.: Alloy: a lightweight object modelling notation. *ACM Trans. Softw. Eng. Methodol.* **11**(2), 256–290 (2002)

Energy-Efficient VM Scheduling in IaaS Clouds

Nguyen Quang-Hung^(✉) and Nam Thoai

Faculty of Computer Science and Engineering, HCMC University of Technology,
VNUHCM, 268 Ly Thuong Kiet Street,
Ho Chi Minh City, Vietnam
{hungnq2,nam}@cse.hcmut.edu.vn

Abstract. This paper investigates the energy-aware virtual machine (VM) scheduling problems in IaaS clouds. Each VM requires multiple resources in fixed time interval and non-preemption. Many previous researches proposed to use a minimum number of physical machines; however, this is not necessarily a good solution to minimize total energy consumption in the VM scheduling with multiple resources, fixed starting time and duration time. We observe that minimizing total energy consumption of physical machines in the scheduling problems is equivalent to minimizing the sum of total busy time of all active physical machines that are homogeneous. Based on these observations, we proposed ETRE algorithm to solve the scheduling problems. The ETRE algorithm's swapping step swaps an allocating VM with a suitable overlapped VM, which is of the same VM type and is allocated on the same physical machine, to minimize total busy time of all physical machines. The ETRE uses resource utilization during executing time period of a physical machine as the evaluation metric, and will then choose a host that minimizes the metric to allocate a new VM. In addition, this work studies some heuristics for sorting the list of virtual machines (e.g., sorting by the earliest starting time, or the longest duration time first, etc.) to allocate VM. Using log-traces in the Feitelson's Parallel Workloads Archive, our simulation results show that the ETRE algorithm could reduce total energy consumption average by 48% compared to power-aware best-fit decreasing (PABFD [6]) and 49% respectively to vector bin-packing norm-based greedy algorithms (VBP-Norm-L1/L2 [15]).

Keywords: IaaS cloud · Virtual machine scheduling · Energy efficiency · Cloud computing · Total busy time · Fixed interval

1 Introduction

Cloud computing, which enables Infrastructure-as-a-Service (IaaS), provides users with computing resources in terms of virtual machines (VMs) to run their applications [4, 5, 10, 14, 18]. Infrastructure of cloud systems are built from virtualized data centers with thousands of high-performance computing servers

[4, 5, 18]. Power consumption in these large-scale data centers requires multiple megawatts [9, 14]. Le et al. [14] estimate the energy cost of a single data center is more than \$15M per year. As these data centers scale, they will consume more energy. Therefore, advanced scheduling techniques for reducing energy consumption of these cloud systems are highly concerned for any cloud providers to reduce energy cost. Increasing energy cost and the need to environmental sustainability address energy efficiency is a hot research topic in cloud systems. Energy-aware scheduling of VMs in IaaS cloud is still challenging [10, 14, 17, 19, 20].

Much previous work [5, 6, 15] showed that the virtual machine allocation is NP-Hard. There are several studies that have been proposed to address the problem of energy-efficient scheduling of VMs in cloud data centers. A number of [5, 6, 15] current techniques for consolidating virtual machines in cloud data centers use bin-packing heuristics (such as First-Fit Decreasing [15], and/or Best-Fit Decreasing [6]). They attempt to minimize the number of running physical machines and to turn off as many idle physical machines as possible. Consider a d -dimensional resource allocation where each user requests a set of virtual machines (VMs). Each VM requires multiple resources (such as CPU, memory, and IO) and a fixed quantity of each resource at a certain time interval. Under this scenario, using a minimum of physical machines may not be a good solution. Our observations show that using a minimum number of physical machines is not necessarily a good solution to minimize total energy consumption. In a homogeneous environment where all physical servers are identical, the power consumption of each physical server is linear to its CPU utilization, i.e., a schedule with longer working time (i.e. total busy time) will consume more energy than another schedule with shorter working time (i.e. total busy time).

Table 1. Example of given six virtual machines (VMs) with their normalized (*) resource demands

VM ID	CPU*	RAM*	Network*	Start-time	Duration (hour)
VM1	0.5	0.1	0.2	0	10
VM2	0.5	0.5	0.2	0	2
VM3	0.2	0.4	0.2	0	1
VM4	0.2	0.4	0.2	0	1
VM5	0.1	0.1	0.1	0	1
VM6	0.5	0.5	0.2	1	9

To the best of our knowledge, our work is the first work that studies increasing time and resource efficiency-based approach to allocate VMs onto physical machines in order that it minimizes the total energy consumption of all physical machines. Each VM requests resource allocation in a fixed starting time and non-preemption for the duration time. We present here an example to demonstrate our ideas to minimize total energy consumption of all physical machines in the

VM placement with fixed starting time and duration time. For example, given six virtual machines (VMs) with their normalized resource demands (CPU*, RAM* and Network* are normalized demand resources to physical server's maximum total capacity resources) described in Table 1. In the example, a bin-packing-based algorithm could result in a schedule S_1 in which two physical machines are used: one for allocating VM1, VM3, VM4, and VM5; and another one for allocating VM2 and VM6. The schedule S_1 has total busy time of the six VMs is $(10 + 9) = 19$ hours. However, in another schedule S_2 in which VMs are placed on three physical machines, VM1 and VM6 on the first physical machine, VM3, VM4 and VM5 on the second physical machine, and VM2 on the third physical machine, then the schedule S_2 has total busy time of the six VMs is only $(10 + 1 + 2) = 13$ hours.

In this paper, we propose a heuristic, namely, ETRE. ETRE heuristic places VMs that request multiple resources in the fixed interval time and non-preemption into physical machines to minimize total energy consumption of physical machines while meeting all resource requirements. Using numerical simulations, we compare the ETRE with the popular modified best-fit decreasing (PABFD) [6], two vector bin-packing norm-based greedy (VBP-Norm-L1/L2) [15], and our previous algorithms (e.g. EPOBF-ST/FT [17], and MinDFT-ST/FT [16]). Using real log-trace (i.e., [1]) in the Feitelson's Parallel Workloads Archive, our simulation results show that the ETRE heuristic with its configurations could reduce total energy consumption average by 48% compared to power-aware best-fit decreasing (PABFD) [6] and 49% respectively to vector bin-packing norm-based greedy algorithms (VBP-Norm-L1/L2 [15]). Additionally, ETRE-ST/LFT/LDTF have also less total energy consumption than our previous heuristics (e.g. MinDFT-ST/FT and EPOBF-ST/FT) in the simulations.

The remainder of the paper is organized as follows. Section 2 describes the energy-aware VM allocation problem with multiple requested resources, fixed starting and duration time. We also formulate the objective of scheduling. The proposed ETRE algorithm is presented in Sect. 3. In Sect. 4 we discuss our performance evaluation using simulations. In Sect. 5 we review the related work. In Sect. 6 we conclude this paper and introduce future work.

2 Problem Description

2.1 Notations

We use the following notations in this paper:

vm_i : The i^{th} virtual machine to be scheduled.

M_j : The j^{th} physical server.

S : A feasible schedule.

P_j^{idle} : Idle power consumption of the M_j .

P_j^{max} : Maximum power consumption of the M_j .

$P_j(t)$: Power consumption of the (M_j) at a time point t .

ts_i : Fixed starting time of vm_i .

dur_i : Duration time of vm_i .

T : Maximum schedule length, which is the time that the last virtual machine will be finished.

$n_j(t)$: Set of indexes of all virtual machines that are assigned to the M_j at time t .

T_j : Total busy time (working time) of the M_j .

e_i : Energy consumption for running the vm_i in the physical machine that the vm_i is allocated.

2.2 Power Consumption Model

In this paper, we use the following energy consumption model proposed in [9] for a physical machine. The power consumption of the M_j , denoted as $P_j(\cdot)$, is formulated as follow:

$$P_j(t) = P_j^{idle} + (P_j^{max} - P_j^{idle})U_j(t) \quad (1)$$

The CPU utilization of the physical server at time t , denoted as $U_j(t)$, is defined as the average percentage of total of allocated computing powers of $n_j(t)$ VMs that is allocated to the M_j . We assume that all cores in CPU are homogeneous, i.e. $\forall c = 1, 2, \dots, PE_j : MIPS_{j,c} = MIPS_{j,1}$, The CPU utilization is formulated as follow:

$$U_j(t) = \left(\frac{1}{PE_j \times MIPS_{j,1}} \right) \sum_{c=1}^{PE_j} \sum_{i \in n_j(t)} mips_{i,c} \quad (2)$$

The energy consumption of the server in a period of $[t_1, t_2]$ is formulated as follow:

$$E_j = \int_{t_1}^{t_2} P_j(U_j(t))dt \quad (3)$$

where:

$U_j(t)$: CPU utilization of the M_j at time t and $0 \leq U_j(t) \leq 1$.

PE_j : Number of processing elements (i.e. cores) of the M_j .

$mips_{i,c}$: Allocated MIPS of the c^{th} processing element to the vm_i by the M_j .

$MIPS_{j,c}$: Maximum capacity computing power (Unit: MIPS) of the c^{th} processing element on the M_j .

2.3 Problem Formulation

Given a set of virtual machines vm_i ($i = 1, 2, \dots, n$) to be scheduled on a set of physical servers M_j ($j = 1, 2, \dots, m$). Each VM is represented as a d-dimensional vector of demand resources, i.e. $vm_i = (x_{i,1}, x_{i,2}, \dots, x_{i,d})$. Similarly, each physical machine is denoted as a d-dimensional vector of capacity

resources, i.e. $M_j = (y_{j,1}, y_{j,2}, \dots, y_{j,d})$. We consider types of resources such as processing element (core), computing power (Million instruction per seconds-MIPS), physical memory (RAM), network bandwidth (BW), and storage. Each vm_i is started at a fixed starting time (ts_i) and is non-preemptive during its duration time (dur_i).

We assume that the power consumption model is linear to CPU utilization. Even if all physical servers are identical and all VMs are identical too, the scheduling is still NP-hard with $d \geq 1$ [15]. With the problem considered in this paper, all physical servers are identical and their power consumption models are linear to their CPU utilization as can be seen in the two Eqs. (1) and (3). The energy consumption of a physical server in a time unit is denoted as E_0 and is the same for all physical servers since the servers are identical. The objective is to find out a feasible schedule S that minimizes the total energy consumption in the Eq. (4) with $i \in \{1, 2, \dots, n\}$, $j \in \{1, 2, \dots, m\}$, $t \in [0; T]$ as following:

$$\text{Minimize } (E_0 \times \sum_{j=1}^m T_j + \sum_{i=1}^n e_i) \quad (4)$$

$$\text{Minimize } (E_0 \times \sum_{j=1}^m T_j + \sum_{i=1}^n e_i) \sim \text{Minimize } (\sum_{j=1}^m T_j) \quad (5)$$

where the total busy time (working time) of a physical server, denoted as T_j , is defined as union of interval times of all VMs that are allocated to a physical machine M_j at time T .

The scheduling problem has the following hard constraints that are described in our previous work [16].

3 Heuristic Based Scheduling Algorithm

In this section, we present our energy-aware scheduling algorithm, namely, ETRE (Energy-aware using increasing Time and Resource Efficiency metric). ETRE presents a performance metric to unify the increasing time and estimated resource efficiency when mapping a new VM onto a physical machine. Then, ETRE will choose a host that has the minimum of the metric. Our previous MinDFT-ST/FT [16] only focused on minimizing the increasing time when mapping a new VM onto a physical machine. The ETRE additionally considers resource efficiency during an execution period of a physical machine in order to fully utilize resources in a physical machine. Furthermore, the core ETRE algorithm can swap an overlapped VM, which has already been assigned to an active physical machine before, with a new VM to minimize total busy time of the physical machine. In this paper, two VMs are overlapped if $ts_1 < ts_2 < (ts_1 + dur_1) < (ts_2 + dur_2)$, where ts_1 , ts_2 , dur_1 , dur_2 are starting times and duration times of two VMs. The core ETRE algorithm will swap a new VM and its overlapped VM together if two VMs meet these conditions: (i) both VMs are of the same VM type (i.e. the same amount of requested resources such

as number of CPU core, physical memory, network bandwidth, storage, etc.); (ii) the new VM has duration time longer than its overlapped VM. Neither our previous MinDFT-ST/FT [16] and the EPOBF-ST/FT [17] have these swapping steps.

Based on Eq. 2, the utilization of a resource r (resource r can be CPU, physical memory, network bandwidth, storage, etc.) of a PM j -th, denoted as $U_{j,r}$, is formulated as follow:

$$U_{j,r} = \sum_{s \in n_j} \frac{V_{s,r}}{H_{j,r}}. \quad (6)$$

where n_j is the list of VMs that are assigned to the physical machine j , $V_{s,r}$ is the amount of the requested resource r of the virtual machine s (note that in our study the value of $V_{s,r}$ is fixed for each user request), and $H_{j,r}$ is the maximum capacity of the resource r in the physical machine j .

Inspired by the work from Microsoft research team [8, 15], resource efficiency of a physical machine j -th, denoted by RE_j , is Norm-based distant [15] of two vectors: normalized resource utilization vector and unit vector $\mathbf{1}$, the resource efficiency is formulated as follow:

$$RE_j = \sum_{r \in \mathcal{R}} ((1 - U_{j,r}) \times w_r)^2 \quad (7)$$

where $\mathcal{R} = \{\text{cpu, ram, netbw, io, storage}\}$: set of resource types in a host, w_r is the weight of resource r in a physical machine.

In this paper, we propose a unified metric for increasing time and resource efficiency that is calculated as:

$$TRE = (t^{diff} \times w_{r=time})^2 + \sum_{r \in \mathcal{R}} ((1 - U_{j,r}) \times w_r)^2 \quad (8)$$

where: t^{diff} is the increasing time (Unit: hours) of total busy time of all physical machines before and after allocating a new VM to this host; and $w_{r=time}$ is weight of time in the TRE metric.

ETRE chooses a physical host that has a minimum value of the TRE metric to allocate a VM. The ETRE can sort the list of VMs by earliest starting time first, or earliest finishing time first, or longest duration time first, etc. The ETRE solves the scheduling problem in the time complexity of $\mathcal{O}(n \times m \times q)$ where n is the number of VMs to be scheduled, m is the number of physical machines, and q is the maximum number of allocated VMs in the physical machines $M_j, \forall j = 1, 2, \dots, m$.

4 Performance Evaluation

4.1 Algorithms

In this section, we study the following VM allocation algorithms:

- PABFD, a power-aware and modified best-fit decreasing heuristic [5,6]. The PABFD sorts the list of VM_i ($i=1, 2, \dots, n$) by their total requested CPU utilization and assigns new VM to any host that has a minimum increase in power consumption.
- VBP-Norm-LX, a family of vector packing heuristics that is presented as Norm-based Greedy with degree $X=1, 2$ [15]. Weights of these Norm-based Greedy heuristics use FFDAvgSum which are $\exp(x)$, which is the value of the exponential function at the point x , where x is the average sum of demand resources (e.g. CPU, memory, storage, network bandwidth, etc.). VBP-Norm-LX assigns a new VM to any host that has minimum of these norm values.
- EPOBF-ST and EPOBF-FT, which is presented in [17], sort the list of VM_i ($i=1, 2, \dots, n$) by their starting time (ts_i) and respectively by their finished time ($ts_i + dur_i$). Both EPOBF-ST and EPOBF-FT choose a host that has the maximum performance-per-watt to assign a new VM. The performance-per-watt is the ratio of the total maximum capacity MIPS and the maximum host's power consumption.
- MinDFT-ST and MinDFT-FT, which is presented in [16], sort the list of VM_i ($i=1, 2, \dots, n$) by their starting time (ts_i) and respectively by their finished time ($ts_i + dur_i$). Both MinDFT-ST and MinDFT-FT allocate each VM (in a given set of VMs) to a host that has a minimum increase in the total completion time of hosts.
- ETRE, our proposed algorithm discussed in Sect. 3. We evaluate the ETRE with some of its configurations: The ETRE-ST sorts the list of virtual machines by VM's earliest starting time first and host's allocated VMs by its finishing times. The finishing time of a virtual machine, which is sum of its starting time and its duration time, is calculated by ($ts_i + dur_i$). The ETRE-LDTF sorts the list of virtual machines by VM's longest duration time first and host's allocated VMs by its finishing time. The ETRE-LFT sorts the list of virtual machines by VM's latest finishing time first and host's allocated VMs by its finishing time.

4.2 Simulated Simulations

We evaluate these algorithms by simulations using the CloudSim [7] to create a simulated cloud data center system that has identical physical machines, heterogeneous VMs, and with thousands of CloudSim's cloudlets [7] (we assume that each HPC job's task is modeled as a cloudlet that is run on a single VM). The information of VMs (and also cloudlets) in these simulated workloads is extracted from a real log-trace (HPC2N Seth log-trace [1]) in Feitelson's Parallel Workloads Archive (PWA) [2] to model HPC jobs. When converted from the log-trace, each cloudlet's length is a product of the system's processing time and CPU rating (we set the CPU rating which is equal to included VM's MIPS). We convert job's submission time, job's start time (if the start time is missing, then the start time is equal to the sum of job's submission time and job's waiting time), job's request run-time, and job's number of processors in job data from the log-trace in the PWA to VM's submission time, starting time and duration

Table 2. Eight (08) VM types in simulations

VM Type	MIPS	Cores	Memory (MBytes)	Net. Bw (Mbits/s)	Storage (GBytes)
Type 1	2500	8	6800	100	1000
Type 2	2500	2	1700	100	422.5
Type 3	3250	8	68400	100	1000
Type 4	3250	4	34200	100	845
Type 5	3250	2	17100	100	422.5
Type 6	2000	4	15000	100	1690
Type 7	2000	2	7500	100	845
Type 8	1000	1	1875	100	211.25

time, and the number of VMs (each VM is created in a round-robin manner with the 8 types of VMs, which can be seen in Table 2 on the number of VMs). Eight types of VMs as presented in the Table 2 are similar to categories in Amazon EC2's VM instances: high-CPU VM, high-memory VM, etc. Fig. 1 shows the chart of starting times and finishing times of the VMs in a simulation (the simulations have the same starting times and duration times of VMs). All physical machines are identical machines. Each physical machine has system information and its power consumption as in Table 3. In the simulations, we use weights as following: (i) the weight of increasing time of mapping a VM to a host: {0.001, 0.01, 1, 100, 3600}; (ii) weights of computing resources such as the number of MIPS per CPU core, physical memory (RAM), network bandwidth, and storage are 940, 24414, 1, 0.0001 respectively. We simulate on the combination of these weights. The total energy consumption of ETRE-ST, ETRE-LFT and ETRE-LDTF are the average of five times simulation with various weights of increasing time (e.g. 0.001, 0.01, 1, 100, or 3600) (Fig. 2).

Table 3. System information of a typical physical machine and its maximum and idle power consumption (P^{max} and P^{idle}) (ratio of P^{idle} and P^{max} is 0.7).

Host Type	MIPS	Cores	Memory (MB)	Network (Mb/s)	Storage (GB)	P^{max}	P^{idle}
M1	3250	16	140084	10000	10000	600	420

We choose PABFD [6] as the baseline algorithm because the PABFD is a famous power-aware best-fit decreasing in the energy-aware scheduling research community. We also compare our proposed VM allocation algorithms with two vector bin-packing algorithms (VBP-Norm-L1/L2) to show the importance of with/without considering VM's starting time and finish time in reducing the total energy consumption of VM placement problem.

Table 4. Result of simulations using the first 400 jobs of the HPC2N Seth log-trace [1]. The normalized energy (Nor. Energy) column is the normalized energy. The energy saving is percentage of energy consumption of an algorithm which reduces in comparison with the energy consumption of PABFD.

Algorithm	#Hosts	#VMs	Energy (KWh)	Nor. Energy	Saving (%)
PABFD	5000	7495	7071.50	1.00	0 %
VBP-Norm-L1	5000	7495	7114.74	1.01	-1 %
VBP-Norm-L2	5000	7495	7114.74	1.01	-1 %
EPOBF-ST	5000	7495	4454.77	0.63	37 %
EPOBF-FT	5000	7495	4409.15	0.62	38 %
MinDFT-ST	5000	7495	4534.36	0.64	36 %
MinDFT-FT	5000	7495	4409.15	0.62	38 %
ETRE-ST	5000	7495	4070.07	0.58	42 %
ETRE-LFT	5000	7495	3418.07	0.48	52 %
ETRE-LDTF	5000	7495	3451.53	0.49	51 %

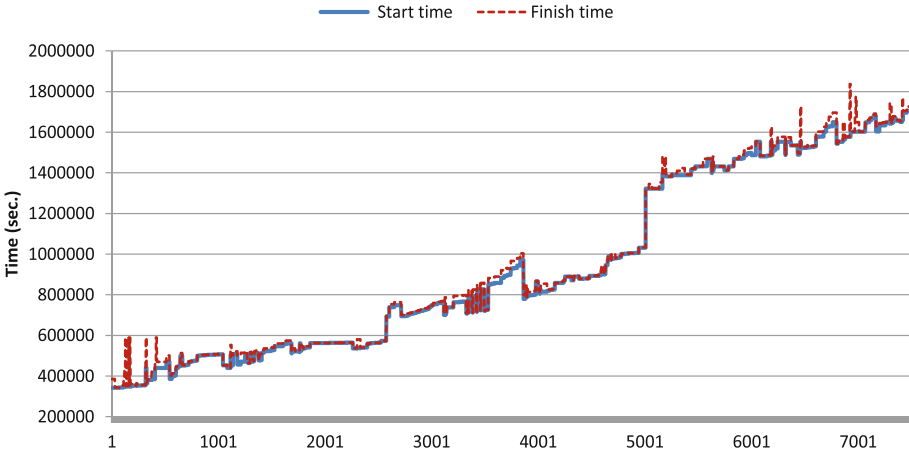


Fig. 1. Starting time (blue line) and finishing time (dotted red line) of VMs in simulations with HPC2N Seth log-trace [1] (Color figure online).

4.3 Results and Discussions

Table 4 shows simulation results of scheduling algorithms solving scheduling problems with 7,495 VMs and 5,000 physical machines (hosts), in which VM's data is converted from the first 400 jobs in the HPC2N Seth log-trace [1]. None of the algorithms use VM migration techniques, and all of them satisfy the Quality of Service (i.e. allocates all resources that user VM requested on-time). We use total energy consumption as the performance metrics for evaluating these VM allocation algorithms. The energy saving shown in both Table 4 is the reduction

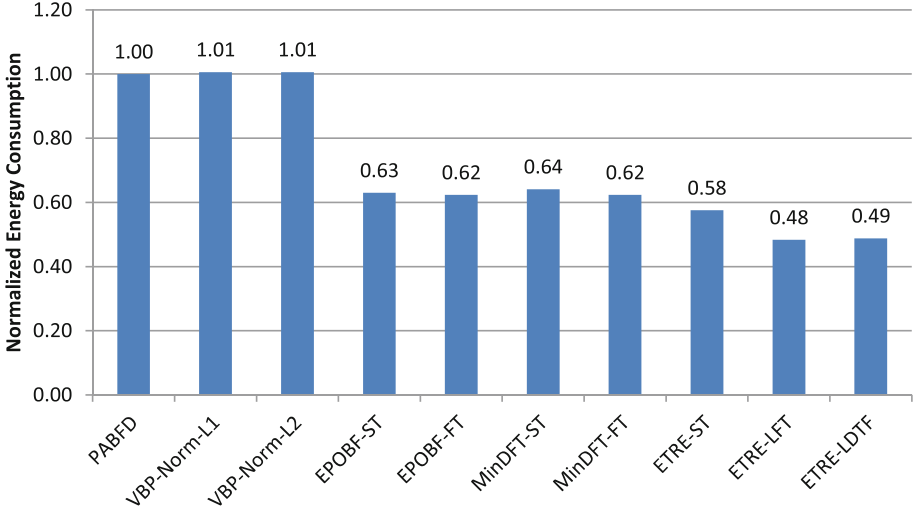


Fig. 2. Normalized energy. Result of simulations with HPC2N Seth log-trace.

of total energy consumption of the corresponding algorithm compared to the baseline PABFD [6] algorithm.

Table 4 shows that our ETRE-ST/LFT/LDTF compared to PABFD [6], can reduce the total energy consumption on the average by 48 %, and 49 % respectively compared to norm-based vector bin-packing algorithms (VBP-Norm-L1/L2) in simulations with the first 400 jobs of the HPC2N Seth log-trace.

The PABFD generates a schedule that uses higher energy consumption than the ETRE-ST/LFT/LDTF because of the following main reasons. First, our hypothesis in this paper is that each VM consumes the same amount of energy in any physical server (e_i) and all physical servers are identical. As a consequence, the PABFD will choose a random physical server to map a new VM. The PABFD sorts the list of VMs by decreasing the requested computing power (e.g. MIPS), therefore the PABFD allocates VMs that firstly have the most requested computing power. In Table 2, all type-3 VMs have the highest requested computing power in the list, the next is a type-1 VM, etc. Instead, our proposed ETRE-ST/LFT/LDTF algorithms assign a new VM to a physical server in such a way that has minimum increase of total busy time of all physical machines and use fully all resources in physical machines.

These ETRE-ST, ETRE-LFT and ETRE-LDTF algorithms perform better than our previous algorithms such as MinDFT-ST/FT and EPOBF-ST/FT in the simulations. Compared to EPOBF-ST and EPOBF-FT, the ETRE-ST, ETRE-LFT and ETRE-LDTF have less total energy consumption on the average by 18 % and 17 % respectively. The ETRE-ST, ETRE-LFT and ETRE-LDTF have also less total energy consumption than the MinDFT-ST and MinDFT-FT on the average by 20 % and 17 %, respectively. In the simulations, swapping between a new VM and its overlapped VM that is allocated to a host reduce

total busy time on the host. For input as in Table 1, the VM2 is removed from the first host, the VM6 will be allocated to the first host.

5 Related Work

Many previous researches [5,6,8,12,19] proposed algorithms that consolidate VMs onto a small set of physical machines (PMs) in virtualized datacenters to minimize energy/power consumption of PMs. Much work has considered the VM placement problem as a bin-packing problem, and have used bin-packing heuristics to place VMs onto a minimum number of PMs to minimize the energy consumption [5,6]. Beloglazov et al. [5,6] have proposed VM allocation problem as bin-packing problem and presented a power-aware best-fit decreasing (denoted as PABFD) heuristic. PABFD sorts all VMs in a decreasing order of CPU utilization and tends to allocate a VM to an active physical server that would take the minimum increase of power consumption. A group in Microsoft Research [15] has studied first-fit decreasing (FFD) based heuristics for vector bin-packing to minimize number of physical servers in the VM allocation problem. Some other work also proposed meta-heuristic algorithms to minimize the number of physical machines. A hill-climbing based allocation of each independent VM is studied in [11]. In the VM allocation problem, however, minimizing the number of used physical machines is not equal to minimizing the total energy consumption of all physical machines.

Takouna et al. [19] presented power-aware multicore scheduling and their VM allocation algorithm selects a host which has the minimum increasing power consumption to assign a new VM. The VM allocation algorithm, however, is similar to the PABFDs [6] except that it concerns memory usage in a period of estimated runtime for estimating the host's energy. The work also presented a method to select optimal operating frequency for a (DVFS-enabled) host and configure the number of virtual cores for VMs. Our proposed ETRE algorithm that is different from these previous work. Our ETRE algorithm uses the VM's fixed starting time and duration time to minimize the total working time on physical servers, and consequently minimize the total energy consumption in all physical servers.

In 2007, Kovalyov et al. [13] presented a work to describe characteristics of a fixed interval scheduling problem in which each job has fixed starting time, fixed processing time, and is only processed in the fixed duration time on a available machine. The scheduling problem can be applied in other domains. Angelelli et al. [3] considered interval scheduling with a resource constraint in parallel identical machines. The authors proved the decision problem is NP-complete if the number of constraint resources in each parallel machine is a fixed number greater than two.

6 Conclusions and Future Work

In this paper, we formulated an energy-aware VM allocation problem with fixed starting time and non-preemption. We also discussed our two key observations

in the VM allocation problem. First, minimizing total energy consumption is equivalent to minimizing the sum of total busy time of all physical machines (PMs). For some possible schedules, which have same ETRE-ST/LFT/LDTF of all PMs, the TRE metric decides a schedule that has higher resource efficiency. Second, swapping between an unallocated VM and its overlapped VM, which has already been allocated to a PM, can reduce the ETRE-ST/LFT/LDTF of all PMs. Based on these observations, we proposed ETRE algorithm to solve the energy-aware VM allocation with fixed starting time and duration time.

Our proposed ETRE can reduce the total energy consumption of the physical servers compared with that of other algorithms in simulation results on the HPC2N Seth [1] in the Feitelson's PWA [2]. The combination of ETRE with its sorting list of virtual machines by latest finishing time first (ETRE-LFT) has the least total energy consumption in these simulations.

In future, we are developing ETRE into a cloud resource management software (e.g. OpenStack Nova Scheduler). We are also working on IaaS cloud systems with heterogeneous physical servers and job requests consisting of multiple VMs. Moreover, we will study how to choose the right weights of time and resources (e.g. computing power, physical memory, network bandwidth, etc.) in another paper.

Acknowledgment. This research was conducted within the “Studying and developing practical heuristics for energy-aware virtual machine-based lease scheduling problems in cloud virtualized data centers” sponsored by TIS, and a fund by HCMUT (under the grant number T-KHMT-2015-33). As an Erasmus Mundus Gate project’s PhD student at The Johannes Kepler University (JKU) Linz, I am thankful to Prof. Dr. Josef Kueng as supervisor. I am also thankful to all reviewers.

References

1. The HPC2N Seth log-trace (HPC2N-2002-2.2-cln.swf.gz file). http://www.cs.huji.ac.il/labs/parallel/workload/1_hpc2n/HPC2N-2002-2.2-cln.swf.gz. Accessed 1 May 2015
2. Feitelson’s Parallel Workloads Archive. <http://www.cs.huji.ac.il/labs/parallel/workload/>. Accessed 31 Januray 2014
3. Angelelli, E., Filippi, C.: On the complexity of interval scheduling with a resource constraint. *Theoret. Comput. Sci.* **412**(29), 3650–3657 (2011). <http://www.sciencedirect.com/science/article/pii/S0304397511002623>
4. Barroso, L.A., Clidaras, J., Hölzle, U.: The datacenter as a computer: an introduction to the design of warehouse-scale machines. *Synth. Lect. Comput. Architect.* **8**(3), 1–154 (2013)
5. Beloglazov, A., Abawajy, J., Buyya, R.: Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing. *Future Gener. Comput. Syst.* **28**(5), 755–768 (2012)
6. Beloglazov, A., Buyya, R.: Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers. *Concurrency Comput. Pract. Exper.* **24**(13), 1397–1420 (2012)

7. Calheiros, R.N., Ranjan, R., Beloglazov, A., De Rose, C.A.F., Buyya, R.: Cloudsim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. *Softw. Pract. Exper.* **41**(1), 23–50 (2011)
8. Chen, L., Shen, H.: Consolidating complementary VMs with spatial/temporal-awareness in cloud datacenters. In: *IEEE INFOCOM 2014 - IEEE Conference on Computer Communications*, pp. 1033–1041. IEEE, April 2014. <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6848033>
9. Fan, X., Weber, W.D., Barroso, L.: Power provisioning for a warehouse-sized computer. In: *ISCA*, pp. 13–23 (2007)
10. Garg, S.K., Yeo, C.S., Anandasivam, A., Buyya, R.: Energy-efficient scheduling of HPC applications in cloud computing environments. CoRR abs/0909.1146 (2009)
11. Goiri, I., Julia, F., Nou, R., Berral, J.L., Guitart, J., Torres, J.: Energy-aware scheduling in virtualized datacenters. In: *2010 IEEE International Conference on Cluster Computing*, pp. 58–67. IEEE, September 2010. <http://doi.ieeecomputersociety.org/10.1109/CLUSTER.2010.15>, <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5600320>
12. Knauth, T., Fetzer, C.: Energy-aware scheduling for infrastructure clouds. In: *4th IEEE International Conference on Cloud Computing Technology and Science Proceedings*, pp. 58–65. IEEE, December 2012, <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6427569>
13. Kovalyov, M.Y., Ng, C., Cheng, T.E.: Fixed interval scheduling: models, applications, computational complexity and algorithms. *Eur. J. Oper. Res.* **178**(2), 331–342 (2007)
14. Le, K., Bianchini, R., Zhang, J., Jaluria, Y., Meng, J., Nguyen, T.D.: Reducing electricity cost through virtual machine placement in high performance computing clouds. In: *SC*, p. 22 (2011)
15. Panigrahy, R., Talwar, K., Uyeda, L., Wieder, U.: Heuristics for vector bin packing. Technical report, Microsoft Research (2011)
16. Quang-Hung, N., Le, D.-K., Thoai, N., Son, N.T.: Heuristics for energy-aware VM allocation in HPC clouds. In: Dang, T.K., Wagner, R., Neuhold, E., Takizawa, M., Küng, J., Thoai, N. (eds.) *FDSE 2014*. LNCS, vol. 8860, pp. 248–261. Springer, Heidelberg (2014)
17. Quang-Hung, N., Thoai, N., Son, N.T.: EPOBF: energy efficient allocation of virtual machines in high performance computing cloud. In: Hameurlain, A., Küng, J., Wagner, R., Thoai, N., Dang, T.K. (eds.) *TLDKS XVI*, LNCS 8960. LNCS, vol. 8960, pp. 71–86. Springer, Heidelberg (2015). http://link.springer.com/10.1007/978-3-662-45947-8_6
18. Sotomayor, B.: Provisioning computational resources using virtual machines and leases. Ph.D. thesis, University of Chicago (2010)
19. Takouna, I., Dawoud, W., Meinel, C.: Energy efficient scheduling of HPC-jobs on virtualize clusters using host and VM dynamic configuration. *Oper. Syst. Rev.* **46**(2), 19–27 (2012)
20. Viswanathan, H., Lee, E.K., Rodero, I., Pompili, D., Parashar, M., Gamell, M.: Energy-aware application-centric VM allocation for HPC workloads. In: *IPDPS Workshops*, pp. 890–897 (2011)

Multi-diagram Representation of Enterprise Architecture: Information Visualization Meets Enterprise Information Management

Lam-Son Lê^(✉)

Faculty of Computer Science and Engineering,
HCMC University of Technology, Ho Chi Minh, Vietnam
lam-son.le@alumni.epfl.ch

Abstract. Modeling Enterprise Architecture (EA) requires the representation of multiple views for an enterprise. This could be done by a team of stakeholders having different backgrounds. The enterprise model built by the team consists of a large number of model elements capturing various aspects of the enterprise. To deal with this high complexity, each stakeholder of the team may want to view only a certain aspect of the enterprise model of her interest. Essentially, the stakeholders need a modeling framework for their EA modeling. We devise a visual modeling language and develop a supporting tool called SeamCAD. Instead of managing a list of ill-related diagrams, the tool manages a coherent enterprise model and generates diagrams on demand, i.e. based on the stakeholders' modeling scope and interests. The notation of the SeamCAD language was based on the Unified Modeling Language and comes with distinctive layout for the purposes of visually and explicitly showing hierarchical containment in the diagrams. We also report industrial applications of our tool and language in this paper. We position our work at the intersection of information visualization and enterprise information management.

Keywords: Visual languages · Enterprise modeling · Information visualization · Enterprise architecture · Enterprise information management · UML

1 Introduction

Enterprise Architecture (EA) captures the whole vision of an enterprise in various aspects regarding both business and information technology (IT) resources [1]. In EA, the goal is to align the business resources and IT resources to maintain or improve the competitiveness of the enterprise. EA is a discipline that studies the services offered by an enterprise and its partners to the customer, the services offered by the enterprise to its partners and the organization of the enterprise itself and of its IT. Making an EA project can, for example, help the enterprise gain more customers, reduce the operation costs or increase its agility.

During an EA project, an EA multidisciplinary team of stakeholders having different backgrounds (e.g. marketers, process designers, IT designers, architects) develops an enterprise model that represents the enterprise, its environment and its internals. As illustrated in Fig. 1, the representation of the enterprise include various aspects such as the services offered by the enterprise, the IT systems, as well as their implementation in terms of business processes and IT applications.

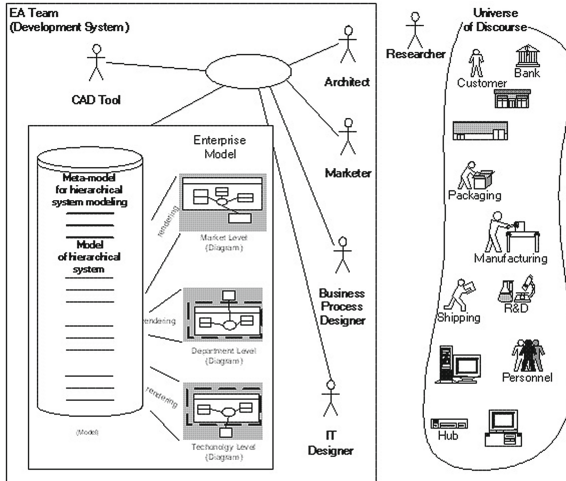


Fig. 1. Stakeholders work with a single, coherent enterprise model to communicate a shared understanding, i.e. the universe of discourse, of the enterprise being modeled.

Modeling is crucial to the success of the EA project as it allows the stakeholders to share their understanding of the enterprise and to document the project. To model the enterprise project, the team develops an agreed and shared representation of the enterprise, its environment and its internals, which all make up an enterprise model. Essentially, the teams need a visual modeling language and a supporting tool that assist them in modeling their enterprise.

Our view on EA is that we can effectively represent an enterprise in terms of hierarchy. Naturally, hierarchical levels represent the enterprise organizational structure. Hierarchical levels also express the granularity of the enterprise model (i.e., going from a coarse-grained description to fine-grained ones). We developed a toolkit and eventually came up with a visual language for modeling EA in a hierarchy-oriented manner. We called this framework SeamCAD. This work was part of (and actually consolidates) an EA methodology called Systemic Enterprise Architecture Methodology (SEAM) [2] that has an established pedigree in the literature and the consulting sector. We discussed the semantics and the modeling constructs of our framework in our previous publications [3, 4]. In this paper, we focus on diagramming techniques, including a cross-discipline notation and tailored layouts, for the purposes of diagrammatically capturing the hierarchical containment in EA.

The rest of this paper is structured as follows. Section 2 discusses the motivation for our work on modeling EA. Section 3 presents the building blocks of SeamCAD and comes up with a meta-model for these building blocks. In Sect. 4 – the core of this paper, we discuss the notation of SeamCAD and its diagramming techniques. Section 5 presents the applicability of SeamCAD. Section 6 surveys related work. Section 7 concludes the paper and points out our future work.

2 Running Example and Motivation

In this section, we argue identify the requirements of visually modeling hierarchical EA by walking through an EA example that requires hierarchical representation and analytically formulating a list of requirements that we need to fulfill.

2.1 Example

Let us consider an example of a bookstore whose management decides to provide the company’s services via the Internet. The management has a goal to specify the services that the bookstore can provide its customers with and to describe how to implement them using business and IT resources. A book-selling market contains a BookValueNetwork and a Customer. The value network consists of three companies: a bookstore company named BookCo (responsible for the service of processing the orders placed by the customer), a shipping company called ShipCo (responsible for shipping the books ordered) and a publishing company PubCo (responsible for supplying the books that were ordered but not yet available in the inventory of the bookstore company). The departmental structure of the bookstore company shows two departments: one for coping with the purchasing data (PurchasingDep) and the other for managing an inventory of books (WarehouseDep). We could have an additional level capturing the IT infrastructure of these departments.

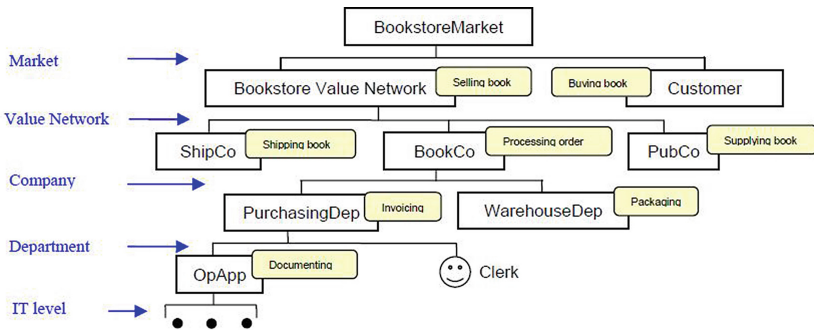


Fig. 2. Initial representation of the enterprise model of the Bookstore example

Figure 2 gives a simplified representation¹ of the organizational structure and services in the enterprise context of the bookstore. In this project, the EA team needs to model the business entities, the IT applications (drawn under regular rectangles in Fig. 2) and their environment, the services offered to the customer by these entities, the company to company (and department to department) business processes, information flow and interaction between the IT system and a clerk who operates it and possibly the system architecture of the IT applications.

2.2 Diagramming Hierarchical EA is Challenging

As exemplified in the Bookstore example presented in Subsect. 2.1, modeling EA involves presenting multiple views of an enterprise that show multiple business entities, IT systems and the services they offer. In our view, we could do this by structuring the model into hierarchical levels (e.g. market level, value network level, company level) each of which can be of interest of particular stakeholders.

Table 1. Four requirements in visually modeling hierarchical EA

Requirement	Brief description
<i>Hierarchy-Oriented</i>	Maintain an explicit organizational level hierarchy and another hierarchy capturing the granularity of the model.
<i>Cross-Discipline Notation</i>	Implement a notation which is systemic, discipline-specific, understandable both by business and UML practitioners;
<i>Coherence & Consistency</i>	Manage a common, coherent enterprise model from which diagrams can be generated based on the stakeholder' modeling scope and interests. These diagrams must be consistent with one another.
<i>Uniformness</i>	To have a uniform diagramming technique for capturing the specification and the implementation of services provided by all business entities and IT systems across hierarchical levels showing the organization of the enterprise.

Developing a visual modeling framework for modeling EA hierarchically that can be used by all stakeholders is challenging. First, hierarchical levels must be made explicit (e.g. from `BookstoreMarket` down to `PurchasingDep` and below). Secondly, the framework should come with a notation that is welcome by stakeholders working with different hierarchical levels (e.g. the market level as well as the IT level) offered by business entities and IT systems and for describing their implementation across hierarchical levels. Thirdly, the modeling framework must

¹ We use an ad-hoc, self-explanatory notation in this initial representation. A regular rectangle represents either a business entity or an IT system. A rounded rectangle can be attached to a regular rectangle to represent the main service offered by the business entity or the IT system drawn under the regular rectangle. The smile symbol stands for individuals. The lines connecting these entities and individuals denote the containment hierarchy. Later on in the paper, we present the UML-based notation of our visual language.

maintain a well-formed enterprise model and the consistency between different diagrams opened by different stakeholders of the team (e.g. *BookCo* appears in multiple views of which one shows the value network level and another shows the company level). Fourthly, the modeling should be done systematically throughout the enterprise model although there might be notational differences from (hierarchical) levels to levels (e.g. modeling techniques used for the value network level, the company level, the department level and the other levels should be the same). We come up with a total of four requirements in Table 1.

3 Definition and Meta-modeling

The SeamCAD visual modeling language was defined following the principles of the SEAM and the modeling concepts of the RM-ODP (Reference Model of Open Distributed Processing) [5]. SEAM is a family of methods for seamless integration between disciplines. SEAM for Enterprise Architecture is an enterprise architecture method that belongs to the change management category [6]. One of the key principles of SEAM is that EA modeling should be done in a systematic way across all hierarchical levels that are created by perceiving enterprises as hierarchical [2]. In SEAM, all entities are systematically treated either as a whole or as a composite, depending on the view [4]. According to SEAM, an enterprise model has two different hierarchies [2]: organizational level hierarchy (i.e., the organizational structure of the enterprise being represented) and functional level hierarchy (i.e., the granularity of enterprise processes and services). RM-ODP is a standardization effort that defines essential concepts for modeling distributed systems in enterprise context [5]. The standard comes along with a few international standards/recommendations – most notably the recommendation on enterprise languages [7].

The building blocks and meta-modeling of SeamCAD are presented in Subsects. 3.1 and 3.2, respectively.

3.1 Building Blocks and Relations

Table 2 list the building blocks and relations of SeamCAD. The building blocks are defined [3] by systematically applying the SEAM principles and the RM-ODP modeling concepts. The relations explicitly or implicitly put the building blocks in relation in a meaningful way.

Note that the building blocks listed in Table 2 can systematically viewed as a whole or a composite. The building blocks and relation are also systematically used across all hierarchical levels. Our modeling technique directly addresses the fourth requirement in Table 1. An exception does to the human working objects, which are only viewed as a whole². Some blocks are defined with an explicit view (e.g. information objects matter in a working object as a whole, business

² We are of course not interested in representing the internals of human being while modeling EA.

Table 2. Building blocks and relation between them in SeamCAD

Building Block	Definition
Business/IT Working Object	Represents any business unit, IT/software component of the enterprise. We call them business/IT working objects for short and we differentiate them from human working objects (see the next row).
Human Working Object	Represents a human party (could be a single person or a group) of the enterprise. The human working objects differentiated from the business/IT working objects (see the previous row).
Parameter Working Object	Business objects exchanged between business/IT/human working objects during a business collaboration.
Business Collaboration	Interaction between multiple business/IT working objects and/or human working objects. A business collaboration take places within another business/IT working object seen as a composite. This working object is the parent (i.e. belongs to a higher hierarchical level) of the participating working objects.
Information Object	Captures a piece of information that is processed or produced by a business/IT working object (seen as a whole) that is participating in a business collaboration.
Localized Action	Captures an externally-observable action performed by a business/IT working object seen as a whole. It also represents a service offered by the given working object.
Association	UML-like association between two information objects of the same working object.
Generalization	Is-a relation between two information objects or two working objects.
Transition	Captures the order between something happening. The could be (i) relation between two localized actions that are parented by another localized action; (ii) relation between two business collaboration that are parented by another business collaboration.
Start/Stop Transition	Relation coming to (or going from) a localized action (or a business collaboration).
Participation	Relation between a working object and a business collaboration in which it participates.
Composition [∇]	Relation between a model element and its parent model element of the same kind. The model element could be of any building block listed above.
Goal-binding [∇]	Binding of an information object or a localized action to a business collaboration.
Means-binding [∇]	Binding of a business collaboration to a localized action that it implements. The two must be hosted by the same working object.

collaboration appears in a working object as a composite). In Table 2, relations that comes with [∇] are intrinsic relations (i.e., they are needed to make enterprise models meaningful but are not diagrammatically expressed). The ones without this symbol are expressive relations (i.e., they are diagrammatically expressed).

3.2 Meta-model of SeamCAD

Figure 3 depicts SeamCAD enterprise models at a meta level. The building blocks defined in Subsect. 3.1 are represented as UML classes. Note that we introduce

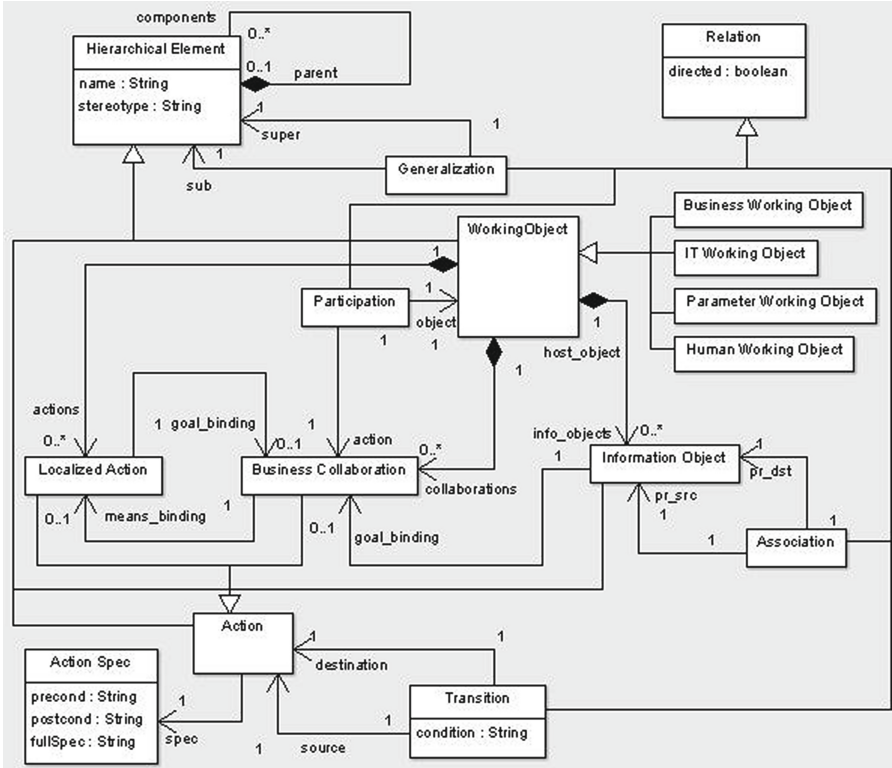


Fig. 3. A UML diagram that describes SeamCAD building blocks and relation between them at a meta level.

abstract classes such as `HierarchicalElement` and `Action` to this meta-model. Although they are not instantiated in SeamCAD enterprise models, they exist in the meta-model to make the meta-model succinct. For the sake of simplicity, we choose not to describe the as-a-whole and as-a-composite views in this meta-model.

4 Notation and Diagramming

The SeamCAD visual language and computer-aided modeling tool were designed to address the requirements identified in Subject. 2.2. In this section, we present how the notation and diagramming techniques, including layout and aesthetics of SeamCAD explicitly fulfill³ the first three requirements.









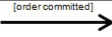

4.1 Notation

As a visual language, SeamCAD has its own notation (hence, we call it the SeamCAD notation) that defines pictograms for building blocks and relation.

³ In Sect. 4.3, we present how the last requirement is met.

The notation also has rules that mandate the way these pictograms are put together in a diagram. To meet the first requirement, namely the *hierarchy-oriented* requirement, pictograms are nested to visually show the containment in the diagrams. More specifically, to visually show that a model element is a component element of another, the pictogram of the latter encloses that of the former. There exists at least an alternative solution to this requirement that is to connect the pictogram of a component model element to that of the parent element using a line. But this way has less expressive power when it comes to visually capturing containment hierarchy in the diagrams. The nested pictograms are employed to diagrammatically represent both the organizational level hierarchy and the functional level hierarchy.

Table 3. Notation of the SeamCAD visual language

Building Block	Pictogram
Business Working Object	
IT Working Object	
Human Working Object	
Parameter Working Object	Same as business or IT working object
Business Collaboration	
Information Object	
Localized Action	
Association	
Generalization	
Transition	
Start/Stop Transition	
Composition	Pictogram of the child element is nested in that of the parent element – a distinctive layout of SeamCAD
Goal-binding	None – not diagrammatically represented
Means-binding	None – not diagrammatically represented

To fulfill the second requirement, namely the *cross-discipline* requirement, most of the SeamCAD pictograms are taken from Unified Modeling Language (UML) – a widely-practiced modeling language for software and system development. Exception goes to the business working objects. Specifically, IT working

objects (i.e., those stand for IT systems or software components), we choose the UML subsystem pictogram as this pictogram conveys two meanings: as a classifier and as a package (in the UML meta-model, the subsystem inherits from both the classifier and the package). As a classifier, it represents something that has both structural and behavioral features. As a package, it can group other model elements, including subsystems, just like a container. For the business working objects, we use a block arrow pictogram that was made popular by Porter in value chain modeling management [8]. In SeamCAD, human working objects are diagrammatically represented under the UML actor pictogram. As such, the SeamCAD visual language has a notation that is understandable to a EA team of cross-discipline stakeholders. The information objects in the SeamCAD visual language takes the UML class pictogram. The localized action in the SeamCAD visual language takes the action pictogram (UML activity diagram). The business collaboration is visually represented under the collaboration pictogram (UML composite structure diagram).

The SeamCAD notation is depicted in Table 3. Note that the depicted pictograms come with exemplary text that describe the names of the model element being represented.

4.2 Layout






In SeamCAD, hierarchical containment (i.e., the composition relationship) is diagrammatically represented by means of diagrammatic layout. As discussed in Subsect. 4.1 (Table 3), the pictogram of the child element is nested in the pictogram of the parent element. Model elements of the enterprise model (which are instantiated from the building blocks of SeamCAD) can be viewed either as a whole or as a composite. The SeamCAD layout is defined as follows. Pictogram for the as-a-composite view fully encloses those of the child elements, which in turn could be viewed either as a whole or as a composite. If model elements of multiple hierarchical levels (could be of the organizational hierarchy, the functional hierarchy or a mixture of both) appear in a diagram, we have multi-level visual containment of pictograms that are rendered in a recursive way.

We use different drawing patterns for the as-a-whole view and the as-a-composite view. Pictograms that represent the as-a-whole view are rendered using solid lines. Exception goes to the pictogram of the business collaboration, which is rendered using dashed lines according to the UML notation. Dotted lines are used for rendering pictograms that represent the as-a-composite view of information objects, business collaboration and localized actions. Together with the visual containment, this difference in terms of drawing patterns would create a sharp contrast between the two views in a diagram. Table 4 illustrates the layout of SeamCAD.

4.3 Generating Diagrams

The third requirement, namely the *coherence & consistency* requirement, is met by the way diagrams are generated. The SeamCAD tool manages a coherent

Table 4. Visual containment in SeamCAD diagrams

Building Block and View	Pictogram
Business Working Object as a Whole	
Business Working Object as a Composite	
Business Collaboration as a Composite	
Information Object as a Composite	
Localized Action as a Composite	

enterprise model. The tool does not explicitly manage a list of diagrams. Instead, it generates diagrammatic representations as a views for a certain portion of the enterprise model being edited. This design explains how we address the challenge of *Well-formedness* by maintaining the consistency between views. The tool provides the user with GUI widgets to specify which portion to be viewed by selecting a business working object from the model. The tool also permits the user to specify the viewing modes (i.e. as-a-whole or as-a-composite) of the selected working object and of relevant business collaborations. Algorithms 1, 2, 3 and 4 define the procedures needed for generating such a view.

The process of generating views starts with an invocation to Algorithm 1 on the main business working object of a view. The procedure will walk through component working objects and business collaborations if the working object is viewed as a composite. If the viewing mode is as-a-whole, it will walk through localized actions and information objects of the selected working object. For each walk-through, depending on the type of the model element being visited, the procedure either recursively calls itself or another procedure.

Algorithms 2, 3 and 4 all have the same patterns: recursively traversing component elements of the same building block to create pictograms according to the SeamCAD notation. Recursive invocations stop when reaching an element that is viewed as a whole.

Viewing modes for component elements can be fetched from the viewing mode of the parent element. This requires a data structure for V_{bwc} , V_{bc} , V_{ia} and V_{io} that mimics the model hierarchies. The third input parameter of Algorithms 3 and 4 stands for the viewing of a collaboration to which the given localized action or information object is connected via a goal binding. This viewing mode may override the viewing mode that is represented by the second input parameter of these two algorithms.

Algorithm 1. buildWorkingObjectGraph

Input: business working object **bwo**; viewing mode V_{bwo} for **bwo**;
Data: A list of pictograms L_{pict} ;

```

1 if  $V_{bwo}$  stands for as-a-composite mode then
2   p ← create a pictogram for working object bwo seen as a composite;
3    $L_{pict} \leftarrow L_{pict} + p$ ;
4   foreach business collaboration bc of bwo do
5      $v_{bc} \leftarrow$  viewing mode for bc, from  $V_{bwo}$ ;
6     buildCollaborationGraph( $L_{pict}$ , bc,  $v_{bc}$ );
7   end
8   foreach component working object wc of bwo do
9      $v_{wc} \leftarrow$  viewing mode for wc, from  $V_{bwo}$ ;
10    buildWorkingObjectGraph( $L_{pict}$ , wc,  $v_{wc}$ );
11  end
12 else
13  p ← create a pictogram for business working object bwo seen as a whole;
14   $L_{pict} \leftarrow L_{pict} + p$ ;
15  foreach localized action la of bwo do
16     $v_{la} \leftarrow$  viewing mode for la, from  $V_{bwo}$ ;
17     $v_{goal} \leftarrow$  viewing mode of some collaboration to which la is connected
    via a goal binding;
18    buildLocalizedActionGraph( $L_{pict}$ , la,  $v_{la}$ ,  $v_{goal}$ );
19  end
20  foreach information object io of bwo do
21     $v_{io} \leftarrow$  viewing mode for io, from  $V_{bwo}$ ;
22     $v_{goal} \leftarrow$  viewing mode of some collaboration to which io is connected
    via a goal binding;
23    buildInformationObjectGraph( $L_{pict}$ , io,  $v_{io}$ ,  $v_{goal}$ );
24  end
25 end

```

All these algorithms are specified as procedures having side effect - adding pictograms into a list (declared in **Data** sections) while visiting model elements. Upon completion, the list is populated with pictograms needed for generating a view according to user's preferences. As such, the life-cycle of this list is beyond the scope of the algorithms (i.e. it should be constructed before calling the algorithms and maintained so long as the view exists). The view is rendered by simply drawing all pictograms in this list. Handlers that respond to user's commands (e.g. selecting/dragging a pictogram by using computer mouse) could straightforwardly be implemented by iterating through this list checking which pictogram is hit by a given mouse event.

Algorithm 2. buildCollaborationGraph

Input: business collaboration bc ; viewing mode V_{bc} for bc ;
Data: A list of pictograms L_{pict} ;

```

1 if  $V_{bc}$  stands for as-a-composite mode then
2    $p \leftarrow$  create a pictogram for business collaboration  $bc$  seen as a composite;
3    $L_{pict} \leftarrow L_{pict} + p$ ;
4   foreach component business collaboration  $cbc$  of  $bc$  do
5      $v_{cbc} \leftarrow$  viewing mode for  $cbc$ , from  $V_{bc}$ ;
6      $buildCollaborationGraph(L_{pict}, cbc, v_{cbc})$ ;
7   end
8 else
9    $p \leftarrow$  create a pictogram for business collaboration  $bc$  seen as a whole;
10   $L_{pict} \leftarrow L_{pict} + p$ ;
11 end

```

Algorithm 3. buildLocalizedActionGraph

Input: information object la ; viewing mode V_{la} for la ; viewing mode V_{goal} ;
Data: A list of pictograms L_{pict} ;

```

1 if  $V_{goal} \neq \text{null}$  then
2   foreach component element  $cla$  of  $la$  do
3      $v_{cla} \leftarrow$  viewing mode for  $cla$ , from  $V_{la}$ ;
4      $v_{cg} \leftarrow$  viewing mode for some collaboration to which  $cla$  is connected to
5     via a goal binding, from  $V_{goal}$ ;
6      $buildLocalizedActionGraph(L_{pict}, cla, v_{cla}, v_{cg})$ ;
7   end
8 else
9   if  $V_{la}$  stands for as-a-composite mode then
10    foreach component information object  $cla$  of  $la$  do
11       $v_{cla} \leftarrow$  viewing mode for  $cla$ , from  $V_{la}$ ;
12       $buildLocalizedActionGraph(L_{pict}, cio, v_{cla}, \text{null})$ ;
13    end
14  end
15   $p \leftarrow$  create a pictogram for localized action  $la$  seen as a whole;
16   $L_{pict} \leftarrow L_{pict} + p$ ;

```

5 Applications

5.1 A Case-Study in a Master's Course on EA and SOA

At the I&C school of the EPFL, Switzerland, a problem-based subject was given to master's students to teach them how to start a manufacturing-and-sale company following a game case-study. Students were asked to make an enterprise model for their imaginary company. They were divided into groups of four to

Algorithm 4. `buildInformationObjectGraph`

Input: information object `io`; viewing mode V_{io} for `io`; viewing mode V_{goal} ;
Data: A list of pictograms L_{pict} ;

```

1 if  $V_{goal} \neq \text{null}$  then
2   foreach component element cio of io do
3      $v_{cio} \leftarrow$  viewing mode for cbc, from  $V_{io}$ ;
4      $v_{cg} \leftarrow$  viewing mode for some collaboration to which cbc is connected to
       via a goal binding, from  $V_{goal}$ ;
5     buildInformationObjectGraph( $L_{pict}$ , cio,  $v_{cio}$ ,  $v_{cg}$ );
6   end
7 else
8   if  $V_{io}$  stands for as-a-composite mode then
9     foreach component information object cio of io do
10       $v_{cio} \leftarrow$  viewing mode for cbc, from  $V_{io}$ ;
11      buildInformationObjectGraph( $L_{pict}$ , cio,  $v_{cio}$ , null);
12    end
13  end
14 end
15 p  $\leftarrow$  create a pictogram for information object io seen as a whole;
16  $L_{pict} \leftarrow L_{pict} + \text{p}$ ;
```

six. Each group represented a company that was to manufacture and sell diesel-powered engines for lightweight aircrafts [9]. An enterprise model of this company was built in a hierarchical manner. All model elements in this enterprise model were expressed using the SeamCAD building blocks and entered in the SeamCAD toolkit. This enterprise model served as a sample model for our students. It has a total of 3 organizational levels and the second level was represented at four different functional levels. The students could refer to it as one of the solutions after having created an enterprise model of their own company.

5.2 Enterprise Model for an ERP-Seeking Company in a Market of Watch Parts Manufacturing

This application was in the context of a Swiss company that is active in the development of ERP (Enterprise Resource Planning) solutions and the research of a new ERP-based method for representing customers' needs in the market of watch manufacturing. A project was set up between this company and our group to develop an enterprise model for organizing, presenting information and doing "savoir-faire" ERP, which is systematically elaborated in the integration and deployment phase of an ERP system by customer companies. In addition, this model should allow the company to analyze the needs of current and future customers, with a goal to manage the technological evolution of the company.

This project led to a master's thesis we supervised [10]. Our master's student developed a SEAM model developed in the project and it was initially made on pieces of paper. This model was partially entered in the SeamCAD tool

resulting in a total of 4 organizational levels of which the third and the fourth were represented at two different functional levels. We entered the most typical portion of their paper-based model in SeamCAD (we did not enter the entire paper-based model due to limited time - the company agreed that the other portions of their model could be entered in similar ways and this labor thus might not bring any additional benefits).

5.3 Designing EA with SEAM and SeamCAD

A project was launched to apply the SEAM method and the SeamCAD toolkit in designing an enterprise model for a project of a new building on the campus of the EPFL, Switzerland. This project led to a master's thesis we supervised [11]. The enterprise model built in this project was useful to specify how the building should be equipped and what IT systems should be installed in the building. It features a total of 3 organizational levels spanning the management of university, the school where the elements of IT equipment were to be installed and the operation of these elements by the school staff/students. In this enterprise model, the second and the third organizational levels were represented at two different functional levels. The model was built using the very first version of SeamCAD has proved its usefulness when showing multiple views that would be of interest to different partners in the project.

6 Related Work

Enterprise-wise information management is becoming increasingly crucial to the success and competitiveness of enterprises. To make the management of information in enterprises more cognitively effective, we need to visually model the enterprise architectures. When it comes to commercial tools for enterprise information modeling, Enterprise Architect⁴ and Mega⁵ are among the most successful products. ArchiMate⁶ is a widely adopted standard for EA modeling [12]. This framework defines three layers in EA – namely, the business layer, the application layer and the technology layer. Each of these layers can further be divided into sub layers. Tools for Archimate are numerous, most notably Archi⁷. Our approach differs primarily in ways notational elements are nested, which truly reflects our view of hierarchical EA.

7 Conclusion

This paper presents our work on modeling EA hierarchically where information visualization meets enterprise information management. An enterprise model

⁴ www.sparxsystems.com/products/ea.

⁵ www.mega.com/en/solution/enterprise-architecture.

⁶ Homepage of Archimate www.archimate.nl.

⁷ www.archimatetool.com.

that represents the enterprise and its environment might include various aspects such as the internal structure of the enterprise and the services provided by the enterprise in question, the business processes and data flow between business entities, the IT components and their interaction. We developed a visual modeling language and its associated tool that are together called SeamCAD.

Future work includes getting more information visualized in the EA models. Work is currently underway to visualize the flow of information objects in EA models. Additionally, in the future versions of SeamCAD, users may express the semantics of business collaboration and localized actions diagrammatically.

References

1. Schekkerman, J.: How to Survive in the Jungle of Enterprise Architecture Framework: Creating or Choosing an Enterprise Architecture Framework. Trafford Publishing, Victoria (2004)
2. Wegmann, A.: On the systemic enterprise architecture methodology (SEAM). In: Proceedings of 5th International Conference on Enterprise Information Systems, Angers, France, pp. 483–490 (2003)
3. Lê, L.S., Wegmann, A.: Hierarchy-oriented modeling of enterprise architecture using reference-model of open distributed processing. *Comput. Stand. Interfaces* **35**(3), 277–293 (2013)
4. Wegmann, A., Lê, L.S., Regev, G., Wood, B.: Enterprise modeling using the foundation concepts of the RM-ODP ISO/ITU standard. *IseB* **5**(4), 397–413 (2007)
5. ISO/IEC 10746-1, 2, 3, 4: ITU-T Recommendation, X.901, X.902, X.903, X.904, Reference Model of Open Distributed Processing. International standard, OMG (1995–1996)
6. Wegmann, A., Regev, G., Rychkova, I., Lê, L.S., de la Cruz, J.D., Julia, P.: Business-IT alignment with SEAM for enterprise architecture. In: Proceedings of 11th IEEE International Conference on Information Reuse and Integration, Annapolis, USA, pp. 111–121. IEEE Computer Society (2007)
7. ISO/IEC: Information Technology - Open Distributed Processing - Reference Model - Enterprise Language — ISO/IEC 15414 — ITU-T Recommendation X.911. SC 7 and ITU (2006)
8. Porter, M.E.: *Competitive Advantage: Creating and Sustaining Superior Performance*, 1st edn. Free Press, New York (1998)
9. Regev, G., Gause, D.C., Wegmann, A.: Experiential learning approach for requirements engineering education. *Requir. Eng.* **14**(4), 269–287 (2009)
10. Dan, D.: ERP Handbook, Outil d'organisation pour l'intégration et le développement de la solution ERP DOPG Prod.com répondant aux besoins actuels et futurs des clients de DOP Gestion SA. Master's thesis, School of Computer and Communication Sciences, EPFL (2008)
11. Langenberg, K.: Designing enterprise architectures with the SEAM method - in-depth study, application and critical analysis. Master's thesis, School of Computer and Communication Sciences, EPFL (2003)
12. Lankhorst, M.: *Enterprise Architecture at Work - Modelling, Communication and Analysis*, 2nd edn. Springer, Heidelberg (2009)

Enhancing the Quality of Medical Image Database Based on Kernels in Bandelet Domain

Nguyen Thanh Binh^(✉)

Faculty of Computer Science and Engineering,
Ho Chi Minh City University of Technology, Ho Chi Minh City, Vietnam
ntbinh@hcmut.edu.vn

Abstract. Diagnostic imaging has contributed significantly to improving the accuracy, timeliness and efficiency of healthcare. Most of medical images have blur combined with noise because of many reasons. This problem will give difficulties to health professionals because each of small details is very useful for the treatment process of doctors. In this paper, we proposed a new method to improve the quality of medical images. The proposed method includes two steps: denoising by Bayesian thresholding in bandelet domain and using the Kernels set for deblurring. We undervested the proposed method by calculating the PSNR and MSE values. This method gives the result better than the other recent methods available in literature.

Keywords: Deblurring · Denoising · Bandelet domain · Bayesian thresholding · Kernels · Medical image

1 Introduction

Modern medicine is based on the diagnosis of clinical diagnosis and subclinical diagnosis. In clinical diagnostics, the diagnostic approach based on images obtained from equipment of which medical machine (diagnostic imaging) plays an important role. Especially, today with the help of the equipment, modern medical machinery and high-tech computer software support make the image clearer and more precise.

The diagnostic imaging is very popular such as X-ray images, ultrasound images, ultrasound - color Doppler, endoscopic image (digestive endoscopy, endoscopic urology, etc.), images computerized tomography Scanner (CT image), magnetic resonance imaging (MRI), etc.

Diagnostic imaging has contributed significantly to improving the accuracy, timeliness and efficiency of diagnosis. As based on ultrasound images, the doctor can accurately measure the relative size of the intra-abdominal solid organs (liver, spleen, kidneys, pancreas, etc.) and detect abnormal masses, if any.

From echocardiography image, the doctor can determine the structure, size of heart chambers, heart valves and major blood vessels. In obstetrics, ultrasound helps determine and monitor the development of the fetus in the womb, CT scanner images help the doctors identify brain pathologies.

Most of medical images have blur combined with noise. There are many reasons to create blur combined with noise in medical images such as the environment, capture device, technician's skills, etc. The medical images which have blur combined with noise will give health professionals difficulties because each of small details is very useful for the treatment process of doctors. The goal of denoising and deblurring is to extrude noise and blur details from the low quality images, but keep edge features.

The images, which have blur or noise, are the difficult problems for image processing. In the past, there were many methods for denoising images, such as: wavelet transform [5–7], contourlet transform [9], nonsubsampled contourlet transform [10, 11], ridgelet transform [12], curvelet transform [13–15], Bayesian framework [19], etc. Most of these methods use the thresholdings [8]: stationary, cycle-spinning, shiftable, steerable wavelet, bayesian thresholding, etc. In case images have blur combined with noise, most of the methods use multilevel thresholdings to remove blur and noise appeared in medical images [22, 24, 27, 28]. Although these methods gave good results and improved the quality of medical images, especially, which is curvelet transform for denoising, the deblurring must depend on the value of point-spread function (PSF).

In this paper, we proposed a method to improve the quality of medical images. The proposed method includes two steps: denoising step by Bayesian thresholding in bandelet domain and using Kernels set for deblurring step. For demonstrating the superiority of the proposed method, we compared the results with the other recent methods available in literature such as: augmented lagrangian [17], Wiener filter. For performance measure, we have used Peak Signal to Noise Ratio (PSNR) and Mean Square Error (MSE) and it has shown that the results of the present method are better than the other methods. The rest of this paper is included: in Sect. 2, we give the basic of bandelet domain, Bayesian thresholding and deblurring image; the proposed method is given clearly in Sect. 3; the experimental and results are presented in Sects. 4 and 5 is our conclusions.

2 Background

2.1 Bandelet Basis

The bandelets [1, 3] have brought optimal approximation results for geometrically regular functions. Bandelets are adapted to geometric boundaries as an orthonormal basis. The bandelets are to perform a transform on functions defined as smooth functions on smoothly bounded domains [1]. The bandelet is an orthogonal, multiscale transform [1, 2, 4]. The bandelet decomposition is applied on orthogonal wavelet coefficients. It is computed with a geometric orthogonal transform. We consider a wavelet transform at a fixed scale 2^j . The wavelet coefficients $\langle f, \psi_{jn} \rangle$ are samples of an underlying regularized function:

$$\langle f, \psi_{jn} \rangle = f * \psi_j(2^{jn})$$

where $\psi_j(x) = \frac{1}{2^j} \psi(-2^{-j}x)$.

The coefficients $\psi_v[n]$ are the coordinates of the bandelet function $b_v \in L^2([0,1]^2)$ in the wavelet basis. A bandelet function [2, 4] is defined by

$$b_v(x) = \sum_n \psi_v[n] \psi_{j_n}(x)$$

Bandelets are as regular as the underlying wavelets. The support of bandelets overlaps in the same way as the support of wavelets overlaps [2]. This is particularly important for reconstructing image approximations with no artifacts. Figure 1 presents a combination of wavelets along a band and its support is thus also along a band.

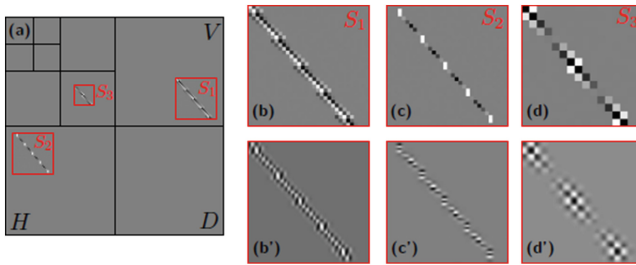


Fig. 1. (a) Localization on the wavelet domain of the squares S_i on which each alpert wavelet vector is defined. (b)–(d) Discrete alpert vectors ψ_{h_i} for various scales 2^l . (b') – (d') Corresponding bandelet functions (source: [2])

On an orthogonal wavelet basis with an orthogonal transformation, we are obtained from bandelets. Apply this transformation to each scale 2^j , an orthogonal basis of $L^2([0,1]^2)$ defined as [2, 4]:

$$B(T) = \overset{def}{\bigcup}_{j \leq 0} \{b_v | \psi_v \in B(T_j)\}, \text{ where } T = \overset{def}{\bigcup}_{j \leq 0} T_j$$

2.2 Bayesian Thresholding

Images are matrixes of numbers which show vital information. Noise appeared in images which is the details are added a noise level. Gaussian noise is popular in noisy image, this is the description for Gaussian noise:

$$w(x, y) = s(x, y) + n(x, y)$$

where, (x, y) is image coordinates, $w(x, y)$ is noisy image, $s(x, y)$ is original image and $n(x, y)$ is noise levels.

With denoising images, we can use many algorithms to remove noise detail from noisy image such as: curvelet transforms give good results, or combine transforms with thresholdings in [12]. But if we use these, we will waste many times for the process.

Most of the existing thresholding procedures are essentially minimax. They do not take into account some specific properties of a concrete object in which we are interested.

If we use many thresholdings, the result image will be over smooth and get the sharpness reduced. The cause of this is that the image must adapt to the value of thresholdings. The idea of wavelet thresholding is divided into hard (T_{hard}) and soft threshold (T_{soft}):

$$T_{\text{hard}}\left(\hat{d}_{jk}, \lambda\right) = \hat{d}_{jk} I\left(\left|\hat{d}_{jk}\right| > \lambda\right) \quad \text{and}$$

$$T_{\text{soft}}\left(\hat{d}_{jk}, \lambda\right) = \text{sign}\left(\hat{d}_{jk}\right) \max\left(0, \left|\hat{d}_{jk}\right| - \lambda\right)$$

where $\lambda \geq 0$ is wavelet coefficients, I is normal coefficients.

Abramovich [14] proposed a Bayesian formalism which gives rise to a type of wavelet threshold estimation in nonparametric regression. They established a relationship between the hyper parameters of the prior model and the parameters of those Besov spaces within which realizations from the prior will fall. The Bayesian threshold solved the standard nonparametric regression problem [18]:

$$y_i = g(t_i) + \epsilon_i, \quad i = 1, \dots, n$$

where $t_i = i/n$ and ϵ_i are independent identically distributed normal variables with zero mean and variance δ^2 , and they will recover the unknown function g from the noised data without assuming any particular parametric forms.

Bayesian thresholdings based on discrete wavelet transforms. The discrete wavelet coefficients are defined by the vector of function values. Based on this vector, we apply them to hard and soft thresholding. In the hard thresholding, the important coefficients remain unchanged while the important coefficients are reduced by the absolute threshold value in the soft thresholding. Meanwhile, the new thresholds are given by equation:

$$d_{jk}^{\text{new}} = T_{\text{hard}}\left(\hat{d}_{jk}, \lambda\right) \quad \text{and}$$

$$d_{jk}^{\text{new}} = T_{\text{soft}}\left(\hat{d}_{jk}, \lambda\right)$$

Based on DWT, the result image after used Bayesian thresholdings is restored by the reverse DWT: $\hat{g} = W^T d^{\text{new}}$.

2.3 Deblurring Images

Most algorithms deblurring based on the value of point-spread function (PSF), then we calculated the approximate value to restore the image. The blur image is described by equation:

$$g = H.f + n$$

where g is blur image, H is the PSF value, f is the original image and n is the noise value.

So the blur image has noise value. Therefore, the deblurring process and denoising process are always parallel.

The deblurring is the process which searches approximation value bases on PSF value. This process is tautologized, again and again, and only stops when the quality of the restoration image reaches close to the original image. Because of the repetition of deblurring image, the time processing is very slow. Most of the deblurring methods based on PSF value, such as: Wiener Filtering and Richardson-Lucy [25], augmented lagrangian method [17].

3 Enhancing the Quality of Medical Image Database

In this section, we propose a method to improve the quality of medical images. As mentioned above, the idea of the proposed method is divided into two steps: denoising and deblurring. In here, we use bayesian thresholding based on bandelet domain to remove noise details, and apply Kernels to removing the blur details of denoising images. The proposed method is used as Fig. 2.

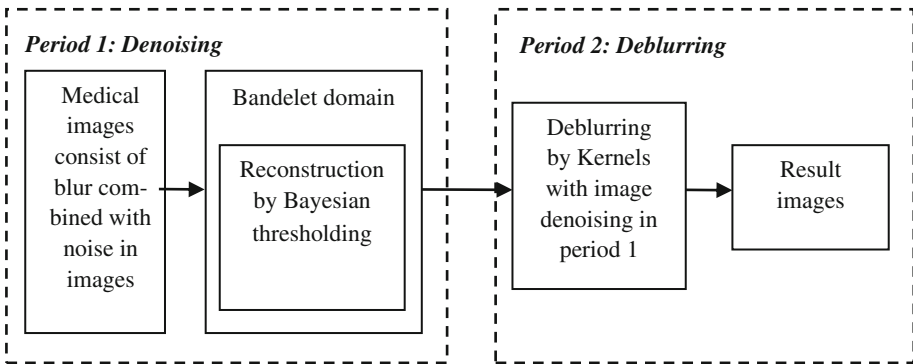


Fig. 2. The processing of proposed method.

3.1 Denoising of Medical Images in Bandelet Domain

In denoising medical image step, the threshold is used. The proposed threshold is complex adaptive in nature. Estimating variance of noise from medical image can use computed threshold. The process of bayesian thresholding based on bandelet domain can be achieved as follows:

- (i) Defining the type of bandelet (filter bank) and the number of scales in the bandelet domain.

- (ii) Doing the bandelet decomposition and calculating sigma_hat. The proposed method uses “db2” for bandelet decomposition. The estimate noise variance σ and signal variance σ_s can be obtained by equation:

$$\sigma = \left(\frac{\text{median}(|w_{i,j}|)}{0.6745} \right)^2$$

$$\sigma_s = \sqrt{\max(\sigma_w^2 - \sigma^2, 0)}$$

$$\text{with } \sigma_w^2 = \frac{1}{n^2} \sum_{i,j=1}^n w^2(i,j)$$

where $w_{i,j}$ is the lowest frequency coefficient after performing transformations.

- (iii) Calculating the thresholds based on sigma_hat.

$$\text{Threshold}_{\text{Bayes}} = \begin{cases} \frac{\sigma^2}{\sigma_s^2}, & \sigma^2 < \sigma_s^2 \\ \max\{|A_m|\}, & \sigma^2 \geq \sigma_s^2 \end{cases}$$

- (iv) Reconstructing the image based on the Bayesian thresholded bandelet coefficients. If the value of pixel detail coefficients is less than thresholding then the result is 0. Else the result is array Y, where each element of Y is 1 if the corresponding element of pixel is greater than zero, 0 if the corresponding element of pixel equals zero, -1 if the corresponding element of pixel is less than zero.

3.2 Deblurring of Denoised Medical Images Based on a Novel Kernels Set

As mentioned above, most of the recent method depended on the PSF values for deblurring images. With the new novel Kernels set [25, 29], deblurring medical image does not require to determine the PSF.

Novel Kernels set is a mathematical operation by equation [25]:

$$R = D \otimes K$$

where D is the degraded image, the Kernel is K, the convolution process is \otimes , and R is the restored image.

In here, we use 20 levels of Kernels, and each of the level is the matrix 3×3 . Using Kernel level depends on blur levels to appear in the medical images. Specially, this method does not depend on the PSF value. We calculate kernel initialization, Kernel estimate and sharp edge gradient map. Depending on this map and the PSF value, we remove blur detail from the blur medical images. In fact, this is a processing to use Laplacian filter, such as Gaussian filter or convolution filter. The old kernels are matrix: {0-1 0, -1 4 -1, 0-1 0} or {-1-1 -1, -1 8 -1, -1-1 -1}.

The result images of denoising step will be the input of deblurring step. In this step, we use the new Kernels set to remove blur out of denoising image by apply matrix 3×3 : $\{0-1 \ 0, -1 \ 5 \ -1, 0-1 \ 0\}$. With this idea, we only use filter, but do not calculate the estimation, initialization and approximation values based on PSF values. So, the time deblurring process is not much. This matrix is similar to masking that deblur.

The output images of this step are removed noise and blur. Then, we undervest the results by the PSNR and MSE values and compare these values with the values from other recent methods which are given clearly in next section.

4 Experiments and Results

We have given clearly our method in Sect. 3, and our results are presented in this section. For performance evaluation, we compare the results of the proposed method with the methods: augmented lagrangian method (ALM) [17], Wiener filter. Our dataset is medical images which are collected in many hospitals. This dataset includes more than 1000 medical images of different sizes: 256×256 , 512×512 . All of the above methods are done on our program and the same images on the similar scale. Our proposed method is compared with ALM, Wiener filter based on the value of Peak Signal to Noise Ratio (PSNR) and Mean Square Error (MSE). The higher the value of PSNR is, the better it is. The smaller the value of MSE is, the better it is. The equation of PSNR defined as:

$$\text{PSNR} = 20\log_{10}\left(\frac{\text{MAX}_1}{\text{MSE}}\right)$$

where MAX_1 is the maximum pixel value of the image; and MSE is defined as:

$$\text{MSE} = \sqrt{\frac{1}{N \times N} \sum_{i=1}^N \sum_{j=1}^N (x_{i,j} - y_{i,j})^2}$$

where x is the image which has blur and noise, y is the image result and $N \times N$ is the size of image.

We test with pairs: motion blur combined with Gaussian noise, Gaussian blur combined with Gaussian noise. The results are given very clearly. In here, we show some test cases in appendix.

Figures 3 and 4 show the denoising and deblurring of blur and noise image which has Gaussian blur combined with Gaussian noise and motion blur combined with Gaussian noise by our proposed method.

From Figs. 3 and 4, we see that the result of the proposed method – figure (d) is better than the other methods – figure (b) and (c). Because the PSNR value of figure (d) is higher than the PSNR value of ALM in figure (b) and Wiener filter in figure (c). That is, the quality of result image by the proposed method is the best.

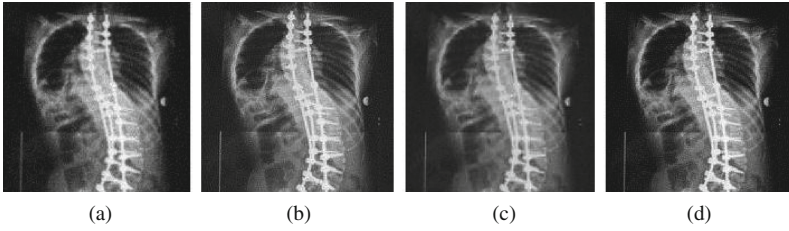


Fig. 3. Denoising and deblurring images in case Gaussian blur combined with Gaussian noise by different methods.

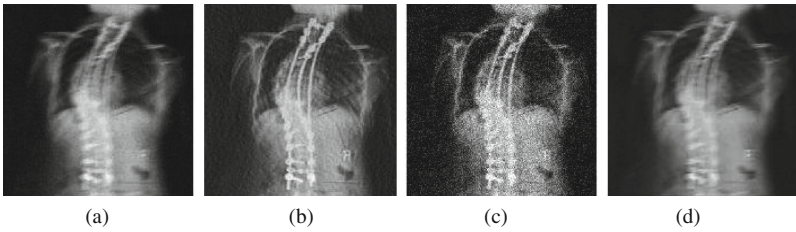


Fig. 4. Denoising and deblurring images in case motion blur combined with Gaussian noise by different methods.

We show the plot of PSNR, MSE values of different methods in denoising and deblurring corrupted in case Gaussian blur combined with Gaussian noise in Figs. 5 and 6, in case motion blur combined with Gaussian noise in Figs. 7 and 8.

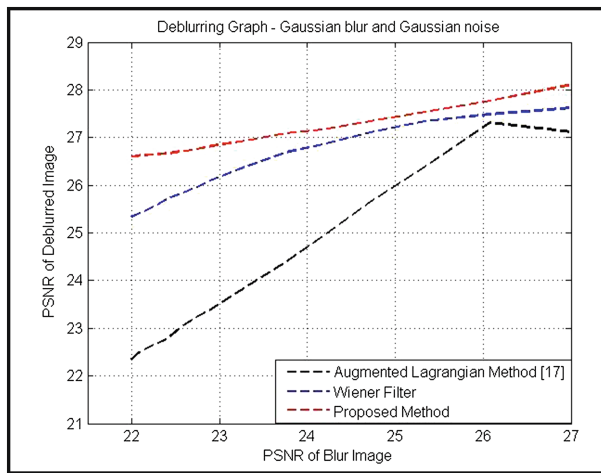


Fig. 5. Plot of PSNR values of denoised and deblurred images in case of Gaussian blur combined with Gaussian noise using different methods.

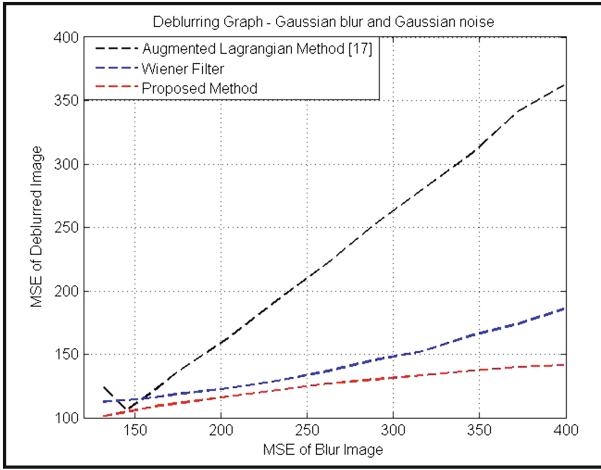


Fig. 6. Plot of MSE values of denoised and deblurred images in case of Gaussian blur combined with Gaussian noise using different methods.

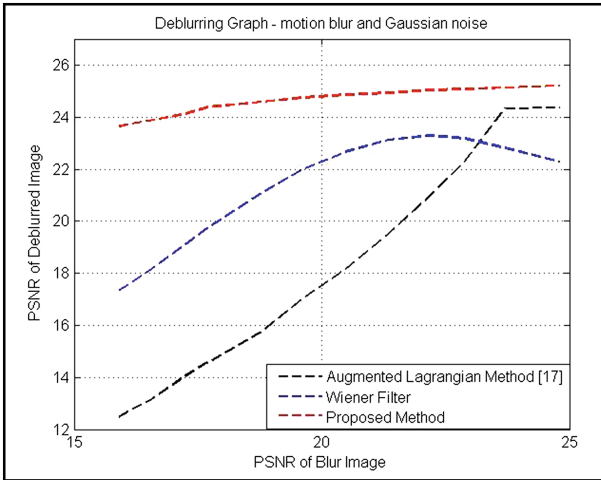


Fig. 7. Plot of PSNR values of denoised and deblurred images in case of motion blur combined with Gaussian noise using different methods.

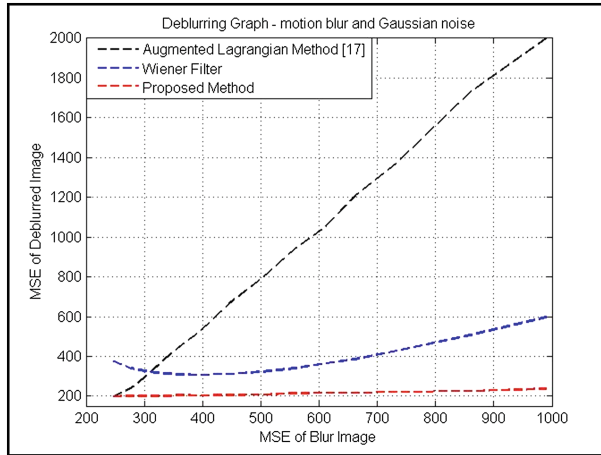


Fig. 8. Plot of MSE values of denoised and deblurred images in case of motion blur combined with Gaussian noise using different methods.

With Figs. 5 and 7, the PSNR values of the proposed method are the highest. In Figs. 6 and 8, the MSE values of the proposed method are the smallest. That means the results of the proposed method improve the quality of medical images which have blur combined with noise. So, the proposed method performs better than augmented lara-gian method and Wiener filter.

5 Conclusions

Improving the quality of medical image sometimes must sacrifice time processing because we need to keep many details. In this paper, we proposed a new method to remove noise and blur in medical images. We divided into two steps: denoising and deblurring. In denoising step, we apply Bayesian thresholding in bandelet domain to remove noise in medical images. Then, we use Kernel matrix to remove blur in medical images. We test the proposed method in cases: motion blur combined with Gaussian noise, Gaussian blur combined with Gaussian noise. The result images compare with ALM [17] and Wiener filter method. Then, we conclude that the proposed method give the better than these methods because we chance the dependence on the PSF value of the other deblurring methods.

Appendix

See Tables [A1](#), [A2](#), [A3](#), [A4](#).

Table A1. PSNR values (dB) of different denoised and deblurred images with Gaussian blur combined with Gaussian noise.

Test image	Image size	Blur and noise image	ALM [17]	Wiener filter	Proposed method
1	256 × 256	19.4320	20.0429	21.6625	21.9052
2		20.5555	20.6731	19.7980	25.8660
3		16.8967	16.8830	19.2017	19.6073
4		20.8832	20.9617	21.0556	26.5889
5		25.6886	26.5646	27.4985	28.0068
6		15.7464	15.7501	16.4946	16.3568
7		20.1199	20.6297	22.0556	22.7812
8		18.8080	19.7501	19.6144	20.5688
9		14.4418	14.6896	15.6200	17.5749
10		21.5807	22.0469	24.5747	25.0719
11		16.2778	17.0291	18.2547	18.3963
12		18.1851	18.7373	19.8721	20.2392
13		15.4203	15.8124	16.9446	17.8252
14		16.4584	16.9531	18.5726	19.0510
15		19.0011	19.4428	21.4829	21.8312
16	512 × 512	21.4479	22.3282	23.0952	23.4989
17		17.8381	18.2437	20.4516	21.0281
18		20.5284	20.6024	23.7115	27.7112
19		21.3950	21.6169	24.7290	27.3587
20		18.4380	18.9251	21.3331	21.5409
21		19.0620	19.5623	21.9663	22.0273
22		30.4161	31.8667	29.1467	31.5594
23		27.4409	28.2649	29.0251	31.5534
24		16.8760	17.2792	18.8454	20.3023
25		18.5067	18.9255	21.4484	21.7973
26		21.7903	22.6140	24.6977	24.8622
27		21.0730	21.6139	24.4481	25.3050
28		22.5920	23.0126	25.8479	26.7079
29		29.0854	30.7627	27.5917	30.7804
30		28.9909	30.3595	27.8331	31.5188

Table A2. PSNR values (dB) of different denoised and deblurred images with motion blur combined with Gaussian noise.

Test image	Image size	Blur and noise image	ALM [17]	Wiener filter	Proposed method
1	256 × 256	16.3033	15.5631	18.0751	17.8975
2		19.8417	17.1131	19.4210	24.5685
3		15.8459	13.2243	17.7172	16.4075
4		19.8586	17.2307	20.1619	24.4456
5		23.2612	23.0414	24.6911	24.8990
6		14.9063	12.4351	16.9841	15.5296
7		17.9669	17.0400	19.5306	19.9939
8		16.9989	16.6669	17.9536	18.3213
9		13.0685	11.1615	14.2849	15.7781
10		19.8910	18.5752	21.7273	22.5824
11		14.4930	13.6713	16.2057	16.3461
12		16.7851	15.3062	18.1975	18.3296
13		14.1962	12.6277	15.6702	16.1290
14		15.3476	13.6305	17.2653	17.3234
15		17.3174	16.1215	19.0947	19.5468
16	512 × 512	19.4334	19.0244	20.6742	20.8923
17		16.0584	13.8915	17.9751	18.4226
18		19.5586	16.7795	21.7706	24.9529
19		20.5329	17.8098	22.0391	25.3056
20		16.5036	15.3094	18.4826	18.8012
21		16.9550	15.9760	18.8860	19.0818
22		28.2117	29.0145	22.4264	28.9309
23		25.8576	24.4650	23.8144	28.7099
24		15.7208	12.9596	17.3331	18.4679
25		17.2849	15.4768	19.4397	19.7994
26		19.8152	18.7848	20.9692	22.0582
27		19.2844	17.8256	21.1117	22.1372
28		20.9622	19.3241	22.5494	23.8660
29		25.7837	27.1080	20.8749	26.6815
30		26.6909	26.6897	21.3499	28.2882

Table A3. MSE values of different denoised and deblurred images with Gaussian blur combined with Gaussian noise.

Test image	Image size	Blur and noise image	ALM [17]	Wiener filter	Proposed method
1	256 × 256	741.0995	643.8544	443.4402	419.3358
2		572.1754	556.8936	681.2099	168.4550
3		1328.6000	1332.9000	781.4644	711.7848
4		530.5941	521.0904	509.9457	142.6231
5		175.4774	143.4222	115.6727	102.8954
6		1731.6000	1730.1000	1430.5000	1504.5000
7		632.5472	562.4807	405.0578	342.7366
8		855.6279	688.7637	710.6176	570.4279
9		2338.3000	2208.6000	1782.7000	1136.6000
10		451.8681	405.8739	226.7841	202.2485
11		1532.2000	1288.7000	971.8781	940.7066
12		987.5644	869.6642	669.6812	615.4046
13		1866.6000	1705.5000	1314.1000	1072.9000
14		1469.8000	1311.5000	903.2673	809.0611
15		818.4069	739.2639	462.1618	426.5397
16	512 × 512	465.8955	380.4200	318.8281	290.5313
17		1069.7000	974.3488	586.0243	513.1752
18		575.7565	566.0359	276.6525	110.1430
19		471.6105	448.1151	218.8666	119.4560
20		931.7088	832.8506	478.3740	456.0246
21		807.0095	719.1955	413.4747	407.7061
22		59.0844	42.3066	79.1420	45.4091
23		117.2175	96.9591	81.3897	45.4721
24		1335.0000	1216.6000	848.2835	606.5284
25		917.0951	832.7709	465.8422	429.8810
26		430.5747	356.1926	220.4483	212.2564
27		507.9037	448.4289	233.4889	191.6820
28		357.9970	324.9559	169.1575	138.7695
29		80.2682	54.5525	113.2164	54.3303
30		82.0341	59.8588	107.0941	45.8355

Table A4. MSE values of different denoised and deblurred images with motion blur combined with Gaussian noise.

Test image	image size	Blur and noise image	ALM [17]	Wiener filter	Proposed method
1	256 × 256	1523.2000	1806.2000	1012.9000	1055.2000
2		674.3944	1264.1000	742.9857	227.1044
3		1692.3000	3094.9000	1099.9000	1487.1000
4		671.7651	1230.3000	626.4577	233.6272
5		306.8774	322.8034	220.7860	210.4660
6		2101.1000	3711.7000	1302.2000	1820.2000
7		1038.5000	1285.5000	724.4724	651.1574
8		1297.7000	1400.9000	1041.7000	957.0936
9		3208.0000	4976.6000	2424.3000	1719.0000
10		666.7816	902.7437	436.8633	358.7930
11		2310.9000	2792.2000	1557.8000	1508.3000
12		1363.2000	1916.3000	984.7548	955.2639
13		2474.3000	3550.6000	1762.2000	1585.6000
14		1898.1000	2818.6000	1220.5000	1204.3000
15		1206.0000	1588.3000	800.9537	721.7664
16	512 × 512	740.8588	814.0390	556.7563	529.4784
17		1611.6000	2654.2000	1036.5000	935.0110
18		719.8113	1365.0000	432.5293	207.8678
19		575.1671	1076.7000	406.5991	191.6535
20		1454.5000	1914.9000	922.1815	856.9684
21		1310.9000	1642.4000	840.3929	803.3384
22		98.1545	81.5890	371.9132	83.1751
23		168.7790	232.5855	270.1693	87.5172
24		1741.8000	3289.4000	1201.6000	925.3195
25		1215.0000	1842.5000	739.7900	680.9901
26		678.5092	860.2080	520.1918	404.8147
27		766.7257	1072.8000	503.3950	397.5231
28		521.0300	759.7551	361.5281	266.9789
29		171.6744	126.5560	531.6018	139.6147
30		139.3142	139.3523	476.5247	96.4408

References

1. Le Pennec, E., Mallat, S.: Sparse geometric image representations with bandelets. *IEEE Trans. Image Process.* **15**, 423–438 (2005)
2. Mallat Cmap, S., Peyré Ceremade, G.: Orthogonal bandelet bases for geometric images approximation. *Commun. Pure Appl. Math.* **LXI**, 1173–1212 (2008)
3. Le Pennec, E., Mallat, S.: Bandelet image approximation and compression. *SIAM J. Multiscale Simul.* **4**(3), 992–1039 (2005)
4. Binh, N.T., Tuyet, V.T.H., Vinh, P.C.: Ultrasound images denoising based context awareness in bandelet domain. In: Vinh, P.G., Alagar, V., Vassev, E., Khare, A. (eds.) *ICCASA. LNICST*, vol. 128, pp. 115–124. Springer, Heidelberg (2014)
5. Strang, G.: Wavelets and dilation equations: a brief introduction. *SIAM Rev.* **31**(4), 614–627 (1989)
6. Edwards, T.: *Discrete Wavelet Transforms: Theory and Implementation* (1992)
7. Kociolek, M., Materka, A., Strzelecki, M., Szczypínski, P.: Discrete wavelet transform – derived features for digital image texture analysis. In: *Proceedings of International Conference on Signals and Electronic Systems*, pp. 163–168 (2001)
8. Binh, N.T., Khare, A.: *Image Denoising, Deblurring and Object Tracking, A new Generation wavelet based approach.* LAP LAMBERT Academic Publishing, Zurich (2013)
9. Do, M.N., Vetterli, M.: The contourlet transform: an efficient directional multiresolution image representation. *IEEE Trans. Image Process.* **14**, 2091–2106 (2005)
10. da Cunha, A.L., Zhou, J., Do, M.N.: Nonsubsampled contourlet transform: theory, design, and applications. *IEEE Trans. Image Proc.* **1**, 3089–3101 (2005)
11. da Cunha, A.L., Zhou, J., Do, M.N.: *Nonsubsampled contourlet transform: filter design and applications in denoising* (2006)
12. Candes, J.: *Ridgelets: Theory and Applications.* Stanford University, Stanford (1998)
13. Zhang, B., Fadili, J.M., Starck, J.L.: Wavelets, ridgelets and curvelets for poisson noise removal. *IEEE Trans. Image Process.* **17**, 1093–1108 (2008)
14. Donoho, D.L., Duncan, M.R.: Digital curvelet transform: strategy, implementation and experiments. In: *Proceedings of SPIE*, vol. 4056, pp. 12–29 (2000)
15. Starck, J.L., Candès, E.J., Donoho, D.L.: The curvelet transform for image denoising. *IEEE Trans. Image Process.* **11**, 670–684 (2002)
16. Binh, N.T., Khare, A.: Multilevel threshold based image denoising in curvelet domain. *J. Comput. Sci. Technol.* **25**, 632–640 (2010)
17. Chan, S.H., Khoshabeh, R., Gibson, K.B., Gill, P.E., Nguyen, T.Q.: An augmented Lagrangian method for total variation video restoration. *IEEE Trans. Image Process.* **20**(11), 3097–3111 (2011)
18. Abramovich, F., Sapatinas, T., Silverman, B.W.: Wavelet thresholding via a Bayesian approach. *J. Roy. Stat. Soc. B* **60**, 725–749 (1998)
19. Sitara, K., Remya, S.: Image deblurring in Bayesian framework using template based blur estimation. *Int. J. Multimed. Appl. (IJMA)* **4**(1), 1–17 (2012)
20. Chui, M., Feng, Y., Wang, W., Li, Z., Xu, X.: Image denoising method with adaptive Bayes threshold in nonsubsampled contourlet domain. *American Applied Science Research Institute* (2012)
21. Lina, J.M., Mayrand, M.: Complex daubechies wavelets. *J. Appl. Comput. Harmonic Anal.* **2**, 219–229 (1995)
22. Khare, A., Tiwary, U.S.: A new method for deblurring and denoising of medical images using complex wavelet transform. *Engineering in Medicine and Biology Society*, pp. 1897 – 1900, IEEE (2005)

23. Candes, J., Demanet, L., Donoho, D.L., Ying, L.: Fast discrete curvelet transforms. *Multiscale Model. Simul.* **5**(3), 861–899 (2006)
24. Khare, A., Tiwary, U.S.: Symmetric daubechies complex wavelet transform and its application to denoising and deblurring. *WSEAS Trans. Signal Process.* **2**, 738–745 (2006)
25. Al-Ameen, Z., Sulong, G., Johar, M.G.M.: Fast deblurring method for computed tomography medical images using a novel kernels set. *Int. J. Bio-Sci. Bio-Technol.* **4**(3), 9–20 (2012)
26. Zhang, W., Yu, F., Guo, H.: Improved adaptive wavelet threshold for image denoising. In: *Control and Decision Conference*, pp. 5958–5963, Chinese (2009)
27. Binh, N.T., Tuyet, V.T.H., Vinh, P.C.: Increasing the quality of medical images based on the combination of filters in ridgelet domain. In: Vinh, P.C., Vassev, E., Hinchey, M. (eds.) *ICTCC 2014. LNICST*, vol. 144, pp. 320–331. Springer, Heidelberg (2015)
28. Tuyet, V.T.H., Binh, N.T.: Reducing impurities in medical images based on curvelet domain. In: Vinh, P.C., Vassev, E., Hinchey, M. (eds.) *ICTCC 2014. LNICST*, vol. 144, pp. 306–319. Springer, Heidelberg (2015)
29. Xu, L., Jia, J.: Two-phase kernel estimation for robust motion deblurring. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) *ECCV 2010, Part I. LNCS*, vol. 6311, pp. 157–170. Springer, Heidelberg (2010)

Information Systems Success: A Literature Review

Thanh D. Nguyen^{1,2(✉)}, Tuan M. Nguyen², and Thi H. Cao³

¹ Banking University of Ho Chi Minh City, Ho Chi Minh City, Vietnam
thanhnd@buh.edu.vn

² HCMC University of Technology, Ho Chi Minh City, Vietnam
n.m.tuan@hcmut.edu.vn

³ Saigon Technology University, Ho Chi Minh City, Vietnam
thi.caohao@stu.edu.vn

Abstract. Information systems (IS) success is a significant topic of interest, not only for scholars and practitioners but also for managers. This paper reviews the IS success research with a multidimensional approach. Various articles in academic journals and international conference on the same theme and between 1992 and 2015 were investigated. The finding indicates that (i) methodological, empirical studies are dominant, (ii) the notion “success” is chiefly represented through individual (e.g., users/customers) benefits, and (iii) DeLone & McLean model is heavily employed during the time. Some research avenues are discussed. Besides, the research gaps will be opportunities for adding development and research trends.

Keywords: Delone & McLean · Information systems success · Literature review

1 Introduction

The worldwide development of information technology (IT) is becoming ever more powerful. However, the majority of the information systems (IS) projects is not successful. According to statistics, in the USA, only about 62 % of IS projects were considered successful [79]. Simultaneously, the Vietnam Government sets two targets which there are having approximately 1 million human resources and increasing the ratio of Internet users in the population up to 55 %–60 % by the year of 2020¹. In the past ten years, the growth rate of IT industry increased by 20 %–25 % on average and is also expected to rise to 30 % by the year of 2020 [9]. Moreover, IT is also the shortest path for developing country. Nevertheless, the projects related to IS are typical more failure that compared with other projects. There are several IS projects that have not achieved the desired objectives in many organizations (e.g., computerising government administrative project – scheme 112; ERP implementation projects of these companies such as Tan Hiep Phat group, Hoang Anh Gia Lai group, Coca-Cola Vietnam; Saigon

¹ Putting Vietnam to become a strong country in ITC project (755/2010 QĐ-TTg), and national telecommunications development plan by the year of 2020 (32/2012 QĐ-TTg) of Vietnam Government.

bank's core banking project). Currently, there are not any statistical evaluation about the main reasons for the failures. The measurement of IS success remains a top concern for researchers, practitioners, and managers [7]. Consequently, the studies on the IS success are essential, it is evidenced by different researches that several models have been proposed to determine and measure the IS success (e.g., [13–15, 20, 47, 71]).

The objective of this study is to present and review the research on the IS success. Specifically, the study answers these research questions: (i) which approaches to appraise the IS success are found in the literature review, (ii) which approaches on IS success are applied in related works, and (iii) what the aggregate results of IS success are in related research.

The study conducts a literature review from the scientific articles which have been published concerning the IS success, by a structured literature review approach. Accordingly, this study analyzes and synthesizes the papers from the international scientific journals and international conferences to provide a comprehensive overview of related research in IS success. Besides, a literature review creates a firm foundation for advancing knowledge, abolishing areas where there is a superfluity of existing literature, and outspreading places where research is needed. The research results will help not only to obtain an overview of the IS success, but also to update the latest publications on IS success. This study has four content items: (1) the research problem is introduced. (2) Literature review, the theoretical foundation for the literature review by defining the terms is implemented, as well as previous researches and widely accepted contributions in IS success are presented. The approach for examining and analyzing the relevant studies on the IS success are outlined. (3) The literature review is presented in the research results. (4) The results, the main contributions, the limitations, and the future works are concluded.

2 Literature Review

2.1 Theoretical Foundation

In 1980, Keen [37] referred to the lack of the scientific basis in IS research and argued that mandatory variables (e.g., user satisfaction, usage) would continue to mislead researchers and dodge the information theory issue. In searching for the IS success, there are many studies have been shown. This is understandable when considers as “*information*”, an output of information systems or a message in communication systems, can be viewed at different levels (technical level, semantic level, and effectiveness level) [13]. In communication context, Shannon & Weaver [72] defined technical level as the propriety and efficiency of the system that effectiveness the information; semantic level as the intended the information in promulgate the intended meaning; and effectiveness level as the effect of the information to the receiver. Based on this basis, Mason [44] considered “*effectiveness*” as “*influence*” and defined information influence level as “*hierarchy of events which take place at the receiving end of an information system which may be used to identify the various approaches that might be used to measure output at the influence level*” [44, p. 227]. According to DeLone & McLean [13], the influence events include the receipt of the information, and the

application of the information, leading to a change in recipient behavior and a change in system performance.

The IS review shows the variety of definitions of IS success, Table 1 provides some representative definitions of IS success. Accordingly, there are not ultimate IS success definitions. Each kind of stakeholders review the IS success in an organization has a different definition [31]. For example, from the developer’s perspective, the IS success is completed on time, under budget, functions correctly. Moreover, customers/users can find an IS successful if it improves user satisfaction or performance [32]. Other side, from the organizational perspective, IS success contributes to the company’s profits or creates the competitive advantage. In addition, IS success also depends on the system type to be evaluated [69].

Table 1. Some representative definitions of IS success

Authors	Definition
Bailey & Pearson [3, p. 530]	<i>“Measuring and analyzing computer user satisfaction is motivated by management’s desire to improve”</i>
Byrd et al. [8, p. 448]	<i>“... the effects of IS along a path can lead to better organizational performance, in this case, lower overall costs”</i>
Gatian [24, p. 119]	<i>“If an effective system is defined as one that adds value to the firm, any measure of system effectiveness should reflect some positive change in user behavior, e.g., improved productivity, fewer errors or better decision making”</i>
Goodhue & Thompson [29, p. 213]	<i>“... MIS success ultimately corresponds to what DeLone & McLean label individual impact or organizational impact. For our purposes, the paper focuses on individual performance impacts as the dependent variable of interest”</i>
Lucas [41, p. 29]	<i>“Because of the extreme difficulty of measuring implementation success through cost/benefit studies, some other indicator of success is needed. The most appealing indicator for this purpose from a measurement standpoint is system use”</i>
Rainer & Watson [61, p. 84]	<i>“An EIS should be developed in response to a specific business need, such as a need to be more responsive to changing customer desires, to improve product quality, or to improve organizational communications. Systems that do not support business objectives are unlikely to succeed”</i>

Generally, “IS success is an IS theory that seeks to provide a comprehensive understanding of IS success by identifying, describing, explaining the relationships among the most critical dimensions of success along which IS are commonly evaluated”. Initial development of the theory was undertaken by DeLone & McLean [13], and was further updated by the original authors a decade later in response to feedback received from other scholars working in IS [14, 15]. Currently, the IS success model has been cited in thousands of scientific papers, and is considered as the most influential theories in contemporary IS research.

DeLone & McLean [13] shows 3 levels of Shannon and Weaver [72]’s information, together with Mason [44]’s expansion of the effectiveness or influence level, 6 IS

categories (Fig. 1). They are system quality, information quality, use, user satisfaction, individual impact, organizational impact. In which, technical level, IS researchers focus on the desired characteristics of the IS itself which produces as *system quality*. In semantic level, researchers choose the information product for desired characteristics as *information quality*. In influence level, researchers analyze the information product interaction with its recipients, by measuring *use* and *user satisfaction*. Researchers are interested in the influence which the information product has on management decisions as *individual impact*. Some researchers and practitioners concern with the information product effect on organizational performance as *organizational impact* [13]. The multidimensional approach with the interdependence among different levels have been regarded as the standard study of the IS success.

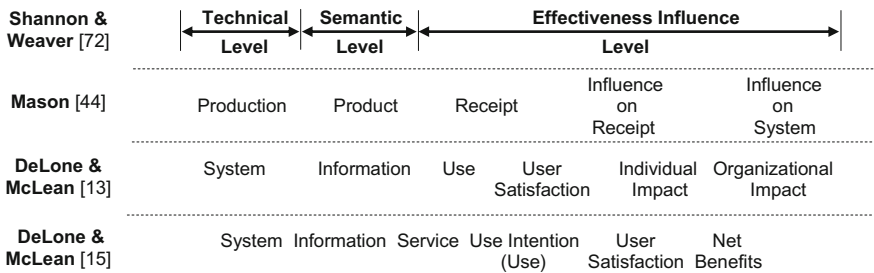


Fig. 1. The categories of IS success

After the publication of the first IS success model (D&M) [13], some scholars claimed that the D&M is incomplete and suggested that more dimensions should be included in the model or proposed the other models. For example, Seddon and Kiew [68]; Seddon [66] argued that the D&M model gaps comprehensiveness and further re-specified the original D&M model by differentiating actual and expected impacts, as well as by incorporating the additional perceived usefulness in Davis’s TAM [12]. Then, Rai et al. [60] showed that both original D&M model and Seddon [66]’s model are adequately explained IS success. Therefore, DeLone & McLean [14, 15] added *service quality* in an updated D&M model. After that, several authors tried to test this model empirically. For example, Gable et al. [20] re-conceptualized the D&M success model and suggested new IS success model. Additionally, Sabherwal et al. [64] conducted a comprehensive analysis to validate the D&M model and highlighted the importance of contextual attributes in IS success. However, this study has been instrumental in synthesizing the quantitative IS success research, it was extended through an inclusive review and analysis of both qualitative and quantitative related studies from 1992 to 2007 by Petter et al. [55]. Further, Petter et al. [57] reviewed research published for the period 1992–2007 and identified the variables that potentially can influence on IS success. Interestingly, Fig. 1 shows 4 approaches as the IS foundational theories, including Shannon & Weaver [72], Mason [44], DeLone & McLean [13], and DeLone & McLean [15].

The literature review shows that some authors use *IS effectiveness* synonymously with *IS success*. Others use IS effectiveness to subsume in *individual impact*;

organizational impact [13], *net benefits* [14, 15]. In this article, the term the IS success is used in the sense of DeLone and McLean’s comprehensive understanding to cover explicitly the whole range of suggested measures. On the other hand, Keen [37] suggests that there are serious gaps in the scientific basis in IS and raises the issue of the dependent variable in IS study. Thus, the scholars have tried to identify additional factors contributing to the IS success. For example, there are many studies on the concept of user satisfaction or usage would continue to evade the information theory problem. Nevertheless, according to Urbach et al. [76], different researchers address the integration aspects of IS success, as well as the integration of different related concepts from the D&M model [13–15]. Although D&M models are still some weaknesses, this model has been become the standard prevail framework in the study of MIS and regularly quoted in the published scientific articles in the top scientific journal [34]. Besides, some authors claim that the D&M model is incomplete, it should add more dimensions, or propose alternative success models [67]. Others focus on the application and accreditation of IS success model [60]. In addition, other domains have been tested using the D&M model that integrated with technology adoption model, including enterprise systems [20, 71], health information systems [84], web-based systems [23], cloud computing [51, 52], social network [50], e-learning [45], e-banking [48, 49], etc.

Ten years after the publication of the D&M model [13], DeLone & McLean [14, 15] proposed an updated IS success model (Fig. 2). The differences between the original and the updated model: (1) adding *service quality* to reflect the service and support importance in e-commerce systems success (e.g., [16, 25, 46]); (2) adding *intention to use* to measure user attitude as a measure of *use* (e.g., [66, 75]); (3) collapsing *individual impact* and *organizational impact* into a parsimonious *net benefits* (e.g., [66]). According to Fig. 2, the updated D&M model consists of 6 IS success factors: 3 quality factors (information, system, and service), intention to to use/use, user satisfaction, and net benefits [14, 15]. Additionally, the updated D&M model can be explained as a system can be evaluated in information terms, system, service quality; these characteristics affect on use intention/use and user satisfaction. Benefits will be achieved by using the IS. Net benefits influence user satisfaction and use of IS.

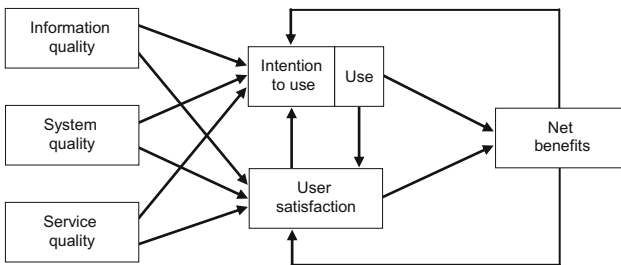


Fig. 2. The updated IS success model (Source: DeLone & McLean [14, 15])

2.2 IS Success Approach

Methodological Review. The literature review is a selection the articles on the scholars research topic, which are presented in a perspective to depict the research problems. Theoretical overview helps share other study results that are closely well-founded about the research being undertaken, appropriating the article continuity, finding the gaps or the widening of previous studies [11, 43]. Besides, the literature review brings research ideas to the scientific outlook, avoid the repetition of the previous study results and conceptual and procedural problems, and gives further problem solving recommendations [18]. Thus, a literature review process is regarded as the scientific procedure that should be guided by the appropriate research methods [43]. A review article is a material evaluation has already been published. Furthermore, evaluating previously published material, the author of a review article searches progress toward clarifying a problem of current research [76].

The basis of this literature review is the relevant articles on the IS success to be researched. Searching scientific resource in a systematic way ensures a number of relatively complete articles related to the topic [42], the IS success in this study.

Identifying a scientific material list that was as comprehensive as possible, specifically, the journals published the conceptual research of the IS success (e.g., [13–15, 66]). According to Webster & Watson [80], the major contributions are likely to be in the top journals, so the top MIS journals are considered. Scopus [65] points out the top journals in the MIS journal rankings. For example, *Australasian Journal of Information Systems (AJIS)*, *Decision Support Systems (DSS)*, *Information Systems Journal (ISJ)*, *Information Systems Research (ISR)*, *Information & Management (I&M)*, *Journal of Association for Information Systems (JAIS)*, *Journal of Management Information Systems (JMIS)*, *MIS Quarterly (MISQ)*, *Omega...* Moreover, the proceedings of the great international conferences on MIS viewed important are also considered [28]. For instance, *Americas Conference on Information Systems (ACIS)*, *Hawaii International Conference on System Sciences (HICSS)*, *International Conference on Information Systems (ICIS)*... In addition, some conceptual definitions of the IS success are shown in Table 1.

For the original D&M model – DeLone & McLean [13] reviewed papers that appeared from 1981 to 1988, and the updated D&M model of IS success – DeLone & McLean [14, 15] literature papers was published between 1992 and 2002. In this study, with the research on IS success after the publication of the original D&M model, the period from 1992 to 2015 (current date) was considered an appropriate period frame selection. However, the literature review relating to the theoretical background in communications or information will be searched at random from the starting point of that theory (e.g., [44, 72]).

Related IS success papers from the scientific resources that have time in defining the period time (1992–2015). In this study, the papers are searched in *Google Scholar* scientific databases with the specific journals and conferences to select articles in the literature review. An initial article list is searched by using the keywords such as “*information systems success*”, “*IS success*”, “*IS effectiveness*”, and “*DeLone & McLean*”

to search for paper titles. To complete the selection process, the paper resulting list is manually reviewed, selecting only the most relevant IS success.

Review Concept and Research Approach. Totally, there are roughly 61 papers have been searched by *Google Scholar* scientific database (scholar.google.com). In this study, the selected literature review is defined an analysis to systematically distribute and describe. The paper consequently examined the classification schemes of similar studies – IS success (e.g., [1, 14, 53, 81]), and adapted evaluation categories that were considered suitable for the literature review. The resulting framework includes some categories: literature review concept (e.g., theoretical foundation), research approach including empirical study (e.g., evaluation perspective, analysis object and unit, data collection and analysis), and non-empirical study (e.g., methodological).

Foundational Theory. Firstly, the list included of the theories on the communications of Shannon & Weaver [72] and information Mason [44] have been considered. Then, the IS success model – D&M model of DeLone & McLean [13], the updated D&M model of DeLone & McLean [14, 15], the technology acceptance model – TAM of Davis [12], and the IS success model of Seddon [66, 67]. Next, others IS success model (e.g., [20, 21, 60]) have also been reviewed. Most of these models are accepted as standard frameworks for IS success measurement.

Research Approach. In this study, the literature review includes conceptual/non-empirical study and empirical study. The conceptual studies are primarily based on ideas rather than on systematic observation (e.g., framework, speculation). They can accommodate some empirical data, but these will be in a secondary role only [76]. On the contrary, the studies are regarded as an empirical study if they apply empirical methods [1] (e.g., survey, interview). In addition, observation of a sampling bias towards empirical studies analyzed in the literature review, also framework/conceptual models, speculation articles were taken into consideration [38]. The research approach of IS success is presented in Table 2.

Table 2. The research approach of IS success

Conceptual study	Empirical study		
Methodological	Evaluation perspective	Analysis object and unit	Data collection and analysis
<ul style="list-style-type: none"> - Framework model - Conceptual model - Speculation - Commentary - Library research - Others 	<ul style="list-style-type: none"> - Users - Top management - IT/IS management - IT/IS team member - External entities - Stakeholders - Others 	<ul style="list-style-type: none"> - IT/IS usage aspect - IT/IS application - IT/IS type - Organizational application - System development methodology aspect - Organizational function - Individual level - Team level - Organizational level - Others 	<ul style="list-style-type: none"> - Survey - Interview - Case study - Experiment - Structural equation modeling (SEM) - Regression analysis - Factor analysis - Variance analysis - Cluster analysis - Others

Methodological. The conceptual studies are classified according to the method. There are three methodological types: framework/conceptual models are researches that intends to classify the framework or conceptual model [84], speculation/commentary are research that is not based on any evidence but glint the author's knowledge or experience [6], and the library are researches that is based on the existing literature review [53, 76]. Others describe other methodological types.

Evaluation Perspective. The evaluation perspective specifies the individual or team in whose interest the IS success evaluation is determined [31]. There are four levels of evaluation perspective, such as users, top management, IT/IS team member, and external entities (e.g., suppliers, customers). Besides, there are two items, called IT/IS executives and stakeholders have been added to evaluation perspective [76]. Different stakeholders in an organization can validly come to different conclusions about the same IS success [69].

Analysis Object and Unit. The analysis object is used to arrange the system type is being examined. According to Seddon et al. [69], analysis object apprehends the following 6 components: IT/IS usage aspect (e.g., user interface), IT/IS application (e.g., core banking), IT/IS type (e.g., management information systems), Organizational or sub-organizational applications, system development methodology aspect, and organizational or sub-organizational IT function.

IS success evaluation should be organized from both a micro and a macro view to build a complete sketch as an analysis unit [31, 76]. Therefore, the IS success should be considered in the multi-level (e.g., individual, team, organization).

Data Collection and Analysis. The data collection contemplates to the research methodology that the scholars exert to pick empirical data [76]. A research methodology analysis offers insights into the study result reliability and generalizability [17]. The research methodology applied to data collection in the empirical papers. For example, survey (e.g., [20, 38]), interview (e.g., [43, 54]), case study (e.g., [18, 27]), experiment (e.g., [23]), etc.

Data analysis techniques consider most commonly used in IS success research, such as structural equation modeling (SEM) (e.g., [5, 82]), regression analysis (e.g., [2]), variance analysis, factor analysis (EFA and CFA) (e.g., [8, 10]). Others used for research using qualitative analysis methods (e.g., [54]).

3 Research Results

There are 56 papers have been searched in the first step, which are considered in journal and conference papers. This literature review is comprehensively estimated the IS success through multidimensional and single dimensional approaches. Therefore, 45 articles remained, the relevant publications that have been analyzed in detail. The research results of the detail analysis show that 14 conceptual/non-empirical or review studies (e.g., [13, 15, 20]) and 26 empirical articles (e.g., [8, 19, 27]). The conceptual/non-empirical and empirical studies of IS success is indicated in Table 3.

Table 3. The conceptual/non-empirical and empirical studies of IS success

Authors	Type	IS success elements								
		SYQ	INQ	SEQ	USI	USE	USS	INI	ORI	NEB
DeLone & McLean 1992 [13]	C	x	x			x	x	x	x	
Seddon & Kiew 1994 [68]	C	x	x				x			
Pitt et al. 1995 [59]	E	x	x			x	x	x	x	
Seddon 1997 [66]	C	x	x				x			x
DeLone & McLean 2002 [14]	C	x	x	x		x	x			x
Rai et al. 2002 [60]	C	x	x			x	x	x		
Briggs et al. 2003 [6]	C									
DeLone & McLean 2003 [15]	C	x	x	x	x	x	x			x
Gable et al. 2003 [20]	E	x	x				x	x	x	
Bharati & Chaudhury 2004 [4]	E	x	x							
DeLone & McLean 2004 [16]	E	x	x	x		x	x			x
Garrity et al. 2005 [23]	E	x					x	x	x	
Sedera et al. 2005 [71]	E	x	x			x		x	x	
Wixom & Todd 2005 [81]	E	x	x		x		x			
Bradley 2006 [5]	E	x	x					x	x	
Byrd et al. 2006 [8]	E	x	x							
Ghandour et al. 2006 [24]	E					x	x			
Sedera 2006 [70]	E		x	x			x			x
Yusof et al. 2006 [84]	C									
Gable et al. 2008 [21]	C	x	x					x	x	
Petter et al. 2008 [55]	R	x	x	x	x	x	x	x	x	x
Petter & McLean 2009 [58]	E	x	x	x	x	x	x			x
Urbach et al. 2009 [76]	R	x	x	x	x	x	x	x	x	x
Floropoulos et al. 2010 [19]	E	x	x	x			x			
Gorla et al. 2010 [30]	R	x	x	x				x		
Urbach et al. 2010 [77]	E	x	x	x		x	x	x	x	
Petter et al. 2012 [56]	R	x	x	x	x	x	x	x	x	x
Urbach & Muller 2012 [78]	E	x	x	x	x	x	x			x
Dorr et al. 2013 [17]	R	x	x	x	x	x	x			x
Koo et al. 2013 [39]	R	x	x				x			
Petter et al. 2013 [57]	R	x	x	x	x	x	x			x
Gao & Bai 2014 [22]	E	x	x				x			
Ghobakhloo et al. 2014 [26]	E	x	x	x		x	x			x
Hsu et al. 2014 [33]	E			x		x				
Kecmanovic et al. 2014 [36]	E						x			x
Lai 2014 [40]	E	x	x				x			x
Tate et al. 2014 [74]	R		x							
Chen et al. 2015 [10]	E		x	x			x			x
Isaias & Issa 2015 [35]	R	x	x	x	x	x	x	x	x	x

(Continued)

Table 3. (Continued)

Authors	Type	IS success elements								
		SYQ	INQ	SEQ	USI	USE	USS	INI	ORI	NEB
Ghobakhloo & Tang 2015 [27]	E	x	x	x		x	x			x
Mohammadi 2015 [45]	E	x	x	x	x	x	x	x		
Rana et al. 2015 [62]	E		x		x		x			
Renzel et al. 2015 [63]	E	x	x				x	x		x
Snead et al. 2015 [73]	E						x			x
Xinli 2015 [82]	E	x	x			x	x			x

C: Conceptual/non-empirical study; E: Empirical study; R: Review study.

SYQ: System quality; INQ: Information quality; SEQ: Service quality; USI: Use intention; USE: Use; USS: User satisfaction; INI: Individual impact; ORI: Organizational impact; NEB: Net benefits.

3.1 Conceptual/Non-empirical Article Results

According to Table 3, the review of all conceptual/non-empirical or review studies paraded are categorized as the type of “C”, including framework/conceptual model, speculation/commentary, and literature review. In which, 16 studies in this review, they are arranged 5 papers as framework/conceptual (e.g., [6, 13, 15, 66]), and 5 papers as speculation/commentary (e.g., [6, 20, 21, 60]). Besides, the literature review studies mustered are categorized as the type of “R”, which have 6 articles (e.g., [35, 56, 57, 76]).

3.2 Empirical Article Results

According to Table 3, the literature review of all empirical studies reviewed are categorized as the type of “E”. In 29 empirical papers, the dominant research analyzes with 19 papers of individual impact (e.g., [20, 21, 45, 59, 62, 77]), and 7 papers of organizational impact (e.g., [20, 21, 23, 77]). Besides, the tested studies as theoretical basis are 9 papers of the original D&M model (e.g., [20, 21, 63, 73, 77]), and 15 papers of the updated D&M model (e.g., [26, 40, 58, 63, 81, 83]).

On the other hand, the analysis object review is used to digest the IS type being evaluated. The empirical studies show 14 papers of the IT/IS application (e.g., [5, 8, 21, 49, 54, 71]). In which, there are 10 papers of IT application is determined, 4 studies evaluate the success of organizational IT/IS application and organizational IT/IS function. Some empirical research validating the framework/conceptual models with group interviews as qualitative method were categorized as others (e.g., [4, 43, 54]). Generally, the research results are presented in Fig. 3.

3.3 Result Discussions

The research results have the circumstantial these findings: (i) several domains are evaluated using the IS success model, enterprise systems or knowledge management systems, have been proposed on the basis of framework/conceptual models. (ii) In 29

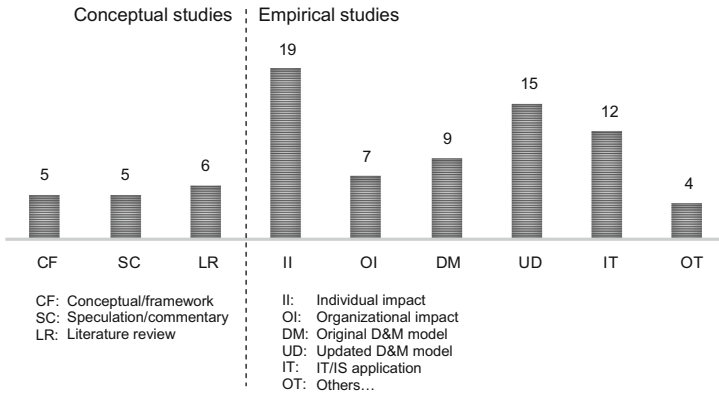


Fig. 3. The summary of research approach of IS success

empirical studies, some studies test the original IS success (D&M) model. Also, most of the studies validate the updated D&M model, integrate with other theoretical as a background for proposing the research models. (iii) The research results show that quantitative analysis is the primary methodology to measure the IS success. The majority of the dominant empirical research is evaluated by users/customers via surveys with the analysis of structural equation modeling (SEM). (iv) Most of the empirical studies appraise IS success at the individual level (19 papers). Only 5 out of 29 empirical papers consider both the individual level and the organizational level of the IS success, these are the gaps for scholars that continue to build more comprehensive research of IS success.

4 Conclusions and Future Work

This study synthesized the IS success and IS success research as a literature review using multidimensional approaches, and classified of articles published between 1992 and 2015 to explore the IS success research. Besides, this research investigated 45 papers in academic journals (e.g., AJIS, DSS, ISJ, ISR, I&M, JAIS, MISQ, Omega...), and international conferences (e.g., ACIS, HICSS, ICIS...) to analyze the theoretical foundation and to research approaches on IS success. In which, the original [13] and the updated [14, 15] IS success models, DeLone & McLean (D&M) model, are the primary theoretical basis of the reviewed empirical studies. This study also provided a comprehensive review of IS success research. Nevertheless, the research gaps will create opportunities for adding development and research trends. This work is still limited in that scientific material sources are not confined all numbers of academic journals and international conferences. Another limitation comprehensively results from the scientific database approach, only *Google Scholar*.

Based on the researched results, the future works have been suggested to extend the material sources. The more meta-analysis of the articles in the literature review will focus on the research classification of IS success. Furthermore, the database-driven approach will be complemented by the manual investigation of content papers.

Acknowledgment. The authors would like to say thank to three anonymous reviewers for their valuable comments on this study.

References

1. Alavi, M., Carlson, P.A.: Review of MIS research and disciplinary development. *J. Manage. Inf. Syst.* **8**(4), 45–62 (1992)
2. Almutairi, H., Subramanian, G.H.: An empirical application of the DeLone and McLean model in the Kuwaiti private sector. *J. Comput. Inf. Syst.* **45**(3), 113–124 (2005)
3. Bailey, J.E., Pearson, S.W.: Development of a tool for measuring and analyzing computer user satisfaction. *Manage. Sci.* **29**(5), 530–545 (1983)
4. Bharati, P., Chaudhury, A.: An empirical investigation of decision-making satisfaction in Web-based decision support systems. *Decis. Support Syst.* **37**(2), 187–197 (2004)
5. Bradley, R., Pridmore, J., Byrd, T.: Information systems success in the context of different corporate cultural types: an empirical investigation. *J. Manage. Inf. Syst.* **23**(2), 267–294 (2006)
6. Briggs, R., Vreede, G., Nunamaker, J., Sprague, R.: Information systems success. *J. Manage. Inf. Syst.* **19**(4), 5–8 (2003)
7. Brynjolfsson, E.: The productivity paradox of information technology. *Commun. ACM* **36**(12), 66–77 (1993)
8. Byrd, T., Thrasher, E., Lang, T., Davidson, N.: A process-oriented perspective of IS success: examining the impact of IS on operational cost. *Omega* **34**(5), 448–460 (2006)
9. Cao, T., Nguyen, H., Truong, C., Ha, H., Nguyen, P.: Forecasting information technology human resource in Ho Chi Minh city from 2011 to 2020. *J. Sci. Technol. Dev.* **14**(2Q), 14–21 (2011)
10. Chen, J., Jubilado, R., Capistrano, E., Yen, D.: Factors affecting online tax filing – an application of the IS success model and trust theory. *Comput. Hum. Behav.* **43**, 251–262 (2015)
11. Cooper, H.: *The Integrative Research Review: A Systematic Approach*. Sage, Thousand Oaks (1984)
12. Davis, F.D.: Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Q.* **13**(3), 319–340 (1989)
13. DeLone, W.H., McLean, E.R.: Information systems success: the quest for the dependent variable. *Inf. Syst. Res.* **3**(1), 60–95 (1992)
14. DeLone, W.H., McLean, E.R.: Information systems success revisited. In: *HICSS Proceedings*. IEEE (2002)
15. DeLone, W.H., McLean, E.R.: Information systems success: a ten-year update. *J. Manage. Inf. Syst.* **19**(4), 9–30 (2003)
16. DeLone, W.H., McLean, E.R.: Measuring e-commerce success: Applying the DeLone & McLean information systems success model. *Int. J. Electron. Commer.* **9**(1), 31–47 (2004)
17. Dorr, S., Walther, S., Eymann, T.: Information systems success—a quantitative literature review and comparison. In: *International Conference on Wirtschaftsinformatik, Leipzig* (2013)
18. Eisenhardt, K.M.: Building theories from case study research. *Acad. Manage. Rev.* **14**(4), 532–555 (1989)
19. Floropoulos, J., Spathis, C., Halvatzis, D., Tshipouridou, M.: Measuring the success of the Greek taxation information system. *Int. J. Inf. Manage.* **30**(1), 47–56 (2010)
20. Gable, G., Sedera, D., Chan, T.: Enterprise systems success: a measurement model. In: *ICIS Proceedings, Seattle* (2003)

21. Gable, G., Sedera, D., Chan, T.: Re-conceptualizing information system success: the IS-impact measurement model. *J. Assoc. Inf. Syst.* **9**(7), 377–408 (2008)
22. Gao, L., Bai, X.: An empirical study on continuance intention of mobile social networking services: integrating the IS success model, network externalities and flow theory. *Asia Pacific J. Mark. Logistics* **26**(2), 168–189 (2014)
23. Garrity, E., Glassberg, B., Kim, Y., Sanders, G., Shin, S.: An experimental investigation of web-based information systems success in the context of electronic commerce. *Decis. Support Syst.* **39**(3), 485–503 (2005)
24. Gatian, A.W.: Is user satisfaction a valid measure of system effectiveness? *Inf. Manage.* **26**(3), 119–131 (1994)
25. Ghandour, A., Deans, K., Benwell, G., Pillai, P.: Measuring eCommerce website success. In: *ACIS Proceedings*, Christchurch (2008)
26. Ghobakhloo, M., Hong, T., Standing, C.: Business-to-business electronic commerce success: a supply network perspective. *J. Organ. Comput. Electron. Commer.* **24**(4), 312–334 (2014)
27. Ghobakhloo, M., Tang, S.H.: Information system success among manufacturing SMEs: case of developing countries. *Inf. Technol. Dev.* **21**(1), 1–28 (2015)
28. Gonzalez, R., Gasco, J., Llopis, J.: Information systems outsourcing: a literature analysis. *Inf. Manage.* **43**(7), 821–834 (2006)
29. Goodhue, D.L., Thompson, R.L.: Task-technology fit and individual performance. *MIS Q.* **19**(2), 213–236 (1995)
30. Gorla, N., Somers, T., Wong, B.: Organizational impact of system quality, information quality, and service quality. *J. Strateg. Inf. Syst.* **19**(3), 207–228 (2010)
31. Grover, V., Jeong, S., Segars, A.: Information systems effectiveness: the construct space and patters of application. *Inf. Manage.* **31**(4), 177–191 (1996)
32. Guimaraes, T., Igbaria, M.: Client/server system success: exploring the human side. *Decis. Sci.* **28**(4), 851–876 (1997)
33. Hsu, M., Chang, C., Chu, K., Lee, Y.: Determinants of repurchase intention in online group-buying: the perspectives of DeLone & McLean IS success model and trust. *Comput. Hum. Behav.* **36**, 234–245 (2014)
34. Hu, P.J.: Evaluating telemedicine systems success: a revised model. In: *HICSS Proceedings*, IEEE (2003)
35. Isaias, P., Issa, T.: Information systems' models for success assessment. In: *High Level Models and Methodologies for Information Systems*, pp. 121–140 Springer, New York (2015)
36. Kecmanovic, D., Kautz, K., Abrahall, R.: Reframing success and failure of information systems: a performative perspective. *MIS Q.* **38**(2), 561–588 (2014)
37. Keen, P.G.: MIS research: reference disciplines and a cumulative tradition. In: *ICIS Proceedings*, Philadelphia (1980)
38. King, W.R., He, J.: External validity in IS survey research. *Commun. Assoc. Inf. Syst.* **16**(1), 45 (2005)
39. Koo, C., Wati, Y., Chung, N.: A study of mobile and internet banking service: applying for IS success model. *Asia Pacific J. Inf. Syst.* **23**(1), 65–86 (2013)
40. Lai, J.Y.: E-SERVCON and E-commerce success: applying the DeLone & McLean model. *J. Organ. End User Comput.* **26**(3), 1–22 (2014)
41. Lucas, H.C.: Empirical evidence for a descriptive model of implementation. *MIS Q.* **2**(2), 27–42 (1978)
42. Machi, L.A., McEvoy, B.T.: *The Literature Review: Six Steps to Success*. Corwin, Thousand Oaks (2012)
43. Marshall, C., Rossman, G.B.: *Designing Qualitative Research*. Sage, Thousand Oaks (2006)

44. Mason, R.O.: Measuring information output: a communication systems approach. *Inf. Manage.* **1**(4), 219–234 (1978)
45. Mohammadi, H.: Investigating users' perspectives on e-learning: an integration of TAM and IS success model. *Comput. Hum. Behav.* **45**, 359–374 (2015)
46. Molla, A., Licker, P.S.: E-commerce systems success: An attempt to extend and respecify the DeLone and McLean model of IS success. *J. Electron. Commer. Res.* **2**(4), 131–141 (2001)
47. Nguyen, T.D.: A structural model for the success of information systems projects. *J. Sci. Technol. Dev.* **18**(2Q), 109–120 (2015)
48. Nguyen, T.D., Cao, T.H.: Proposing the e-banking adoption model in Vietnam. *J. Sci. Technol. Dev.* **14**(2Q), 97–105 (2011)
49. Nguyen, T.D., Cao, T.H.: Structural model for adoption and usage of E-banking in Vietnam. *Econ. Dev. J.* **220**, 116–135 (2014)
50. Nguyen, T.D., Cao, T.H., Tran, N.D.: Structural model for the adoption of online advertising on social network in Vietnam. In ICACCI, pp. 38–43, IEEE (2014)
51. Nguyen, T.D., Nguyen, D.T., Cao, T.H.: Acceptance and use of information system: e-learning based on cloud computing in Vietnam. In: Linawati, Mahendra, M.S., Neuhold, E. J., Tjoa, A.M., You, I. (eds.) *ICT-EurAsia 2014*. LNCS, vol. 8407, pp. 139–149. Springer, Heidelberg (2014)
52. Nguyen, T.D., Nguyen, T.M., Pham, Q.T., Misra, S.: Acceptance and use of e-learning based on cloud computing: the role of consumer innovativeness. In: Murgante, B., Misra, S., Rocha, A.M.A.C., Torre, C., Rocha, J.G., Falcão, M.I., Taniar, D., Apduhan, B.O., Gervasi, O. (eds.) *ICCSA 2014, Part V*. LNCS, vol. 8583, pp. 159–174. Springer, Heidelberg (2014)
53. Palvia, P., Leary, D., Mao, E., Midha, V., Pinjani, P., Salam, A.: Research methodologies in MIS: an update. *Commun. Assoc. Inf. Syst.* **14**(1), 24 (2004)
54. Pare, G., Aubry, D., Lepanto, L., Sicotte, C.: Evaluating PACS success: a multidimensional model. In: *HICSS Proceedings*, IEEE (2005)
55. Petter, S., DeLone, W., McLean, E.: Measuring information systems success: models, dimensions, measures, and interrelationships. *Eur. J. Inf. Syst.* **17**(3), 236–263 (2008)
56. Petter, S., DeLone, W., McLean, E.: The past, present, and future of “IS Success”. *J. Assoc. Inf. Syst.* **13**(5), 341–362 (2012)
57. Petter, S., DeLone, W., McLean, E.: Information systems success: the quest for the independent variables. *J. Manage. Inf. Syst.* **29**(4), 7–62 (2013)
58. Petter, S., McLean, E.: A meta-analytic assessment of the DeLone and McLean IS success model: an examination of IS success at the individual level. *Inf. Manage.* **46**(3), 159–166 (2009)
59. Pitt, L., Watson, R., Kavan, C.: Service quality: a measure of information systems effectiveness. *MIS Q.* **19**(2), 173–187 (1995)
60. Rai, A., Lang, S., Welker, R.: Assessing the validity of IS success models: an empirical test and theoretical analysis. *Inf. Syst. Res.* **13**(1), 50–69 (2002)
61. Rainer, R.K., Watson, H.J.: The keys to executive information system success. *J. Manage. Inf. Syst.* **12**(2), 83–98 (1995)
62. Rana, N., Dwivedi, Y., Williams, M., Lal, B.: Examining the success of the online public grievance redressal systems: an extension of the IS success model. *Inf. Syst. Manage.* **32**(1), 39–59 (2015)
63. Renzel, D., Klamma, R., Jarke, M.: IS success awareness in community-oriented design science research. In: Donnellan, B., Helfert, M., Kenneally, J., VanderMeer, D., Rothenberger, M., Winter, R. (eds.) *DESRIST 2015*. LNCS, vol. 9073, pp. 413–420. Springer, Heidelberg (2015)

64. Sabherwal, R., Jeyaraj, A., Chowa, C.: Information system success: individual and organizational determinants. *Manage. Sci.* **52**(12), 1849–1864 (2006)
65. Scopus: Journal rankings – subject category: management information systems. Scimago Lab (2015). <http://www.scimagojr.com>
66. Seddon, P.B.: A respecification and extension of the DeLone and McLean Model of IS success. *Inf. Syst. Res.* **8**(3), 240–253 (1997)
67. Seddon, P.B.: Implications for strategic IS research of the resource-based theory of the firm: a reflection. *J. Strateg. Inf. Syst.* **23**(4), 257–269 (2014)
68. Seddon, P.B., Kiew, M.: A partial test and development of the DeLone and McLean model of IS success. *ICIS Proc.* **4**(1), 99–110 (1994)
69. Seddon, P.B., Staples, S., Patnayakuni, R., Bowtell, M.: Dimensions of information systems success. *Commun. AIS* **2**(3), 5 (1999)
70. Sedera, D.: An empirical investigation of the salient characteristics of IS-success models. In: *ACIS Proceedings, Acapulco* (2006)
71. Sedera, D., Gable, G., Chan, T.: Measuring enterprise systems success: the importance of a multiple stakeholder perspective. In: *ECIS Proceedings, Turku* (2004)
72. Shannon, E., Weaver, W.: Recent contributions to the mathematical theory of communication. *Math. Theory Commun.* **1**, 1–12 (1949)
73. Snead, K., Magal, S., Christensen, L., Amadi, A.: Attribution theory: a theoretical framework for understanding information systems success. *Syst. Pract. Action Res.* **28**(3), 273–288 (2015)
74. Tate, M., Sedera, D., McLean, E., Jones, A.: Information systems success research: the twenty year update? *Commun. Assoc. Inf. Syst.* **34**(64), 1235–1246 (2014)
75. Taylor, S., Todd, P.A.: Understanding information technology usage: a test of competing models. *Inf. Syst. Res.* **6**(2), 144–176 (1995)
76. Urbach, N., Smolnik, S., Riempp, G.: The state of research on information systems success. *Bus. Inf. Syst. Eng.* **1**(4), 315–325 (2009)
77. Urbach, N., Smolnik, S., Riempp, G.: An empirical investigation of employee portal success. *J. Strateg. Inf. Syst.* **19**(3), 184–206 (2010)
78. Urbach, N., Muller, B.: The updated DeLone and McLean model of information systems success. In: Dwivedi, Y.K., et al. (eds.) *Information Systems Theory. Integrated Series in Information Systems*, vol. 28, pp. 1–18. Springer, New York (2012)
79. Verner, J., Cox, K., Bleistein, S.: Predicting good requirements for in-house development projects. In: *International Symposium on Empirical Software Engineering Proceedings*, pp. 154–163. IEEE (2006)
80. Webster, J., Watson, R.T.: Analyzing the past to prepare for the future: writing a literature review. *MIS Q.* **26**(2), xiii–xxi (2002)
81. Wixom, B.H., Todd, P.A.: A theoretical integration of user satisfaction and technology acceptance. *Inf. Syst. Res.* **16**(1), 85–102 (2005)
82. Xinli, H.: Effectiveness of information technology in reducing corruption in China: a validation of the DeLone and McLean information systems success model. *Electron. Libr.* **33**(1), 52–64 (2015)
83. Yoon, Y., Guimaraes, T., Clevenson, A.: Exploring expert system success factors for business process reengineering. *J. Eng. Technol. Manage.* **15**(2), 179–199 (1998)
84. Yusof, M., Paul, R., Stergioulas, L.: Towards a framework for health information systems evaluation. In: *HICSS Proceedings. IEEE* (2006)

Context-Based Data Analysis and Applications

Facilitating the Design/Evaluation Process of Web-Based Geographic Applications: A Case Study with WINDMash

The Nhan Luong¹(✉), Christophe Marquesuzaa², Patrick Etcheverry²,
Thierry Nodenot², and Sébastien Laborie²

¹ Faculty of Computer Science and Engineering,
Ho Chi Minh City University of Technology,
268 Ly Thuong Kiet Street, District 10, Ho Chi Minh City, Vietnam
nhan@hcmut.edu.vn

² Université de Pau et des Pays de l'Adour, Laboratoire d'informatique,
EA 3000, 64600 Anglet, France
{christophe.marquesuzaa,patrick.etccheverry,thierry.nodenot,
sebastien.laborie}@iutbayonne.univ-pau.fr

Abstract. Web-based geographic applications are continuously evolving and are becoming increasingly widespread. Actually, many Web-based geographic applications have been developed in various domains, such as tourism, education, surveillance and military. However, designing these applications is still a cumbersome task because it requires multiple and high-level technical skills related not only to recent Web technologies but also to technologies dedicated to geographic information systems (GIS). For instance, it requires several components (e.g. maps, multimedia contents, indexing services, databases) that have to be assembled together. Hence, developers have to deal with different technologies and application behaviour models. In order to take the designers out of this complexity and thus facilitate the design/evaluation of Web-based geographic applications, we propose a framework that focuses on both designers' creativity and model executability. This framework has been implemented in a prototype named WINDMash, a Web mashup environment that designers can use both to create and to assess interactive Web-based applications that handle geographical information.

Keywords: Web application generation · Geographical data · Authoring tool · Mashups · Short lifecycle

1 Introduction

Cartography on the Internet has caused a revolution not only in the uses of maps but also in the way to design applications presenting geolocalized data. First research work on geographic information system (GIS) concerned geolocalized

The original version of this paper was revised: The affiliations of the authors were corrected. The erratum to this chapter is available at https://doi.org/10.1007/978-3-319-26135-5_24

data gathering and visualization. A lot of geolocalized data is now available in free geographic databases (geonames.org for example) and new geolocalized data can be easily gathered with a simple smartphone integrating a GPS chip. Moreover, simple tools like Google Maps allow any geolocalized data to be displayed in various formats and on several kinds of maps. Consequently, a fair amount of research and development has been conducted on Web-based application generation thanks to Web 2.0 technologies. Particularly in the domain of geographic information system, specific terms appeared in order to designate Internet Geographic Applications [1]: “GeoWeb”, “Geospatial Web” and “Web Mapping 2.0”. Indeed, many Web-based geographic applications have been developed in different application domains (e.g. tourism, education, surveillance, military) and are using online mapping services (e.g. Google Maps, MapQuest, MultiMap, OpenLayers, Yahoo! Maps or French IGN Geoportail).

However, developing such applications is a cumbersome task. The reasons are twofold:

1. they mix several components (e.g. maps, multimedia contents, indexing services, databases) which have to interact together;
2. developers have to deal with several technologies and different data structures as well as application models.

Hence, design difficulties do not deal anymore with gathering or displaying geolocalized data. Real difficulties now concern the way to design rich applications allowing end-users to interact with this kind of data. Recent research papers [2,3] highlight this problem and the need to improve design methodologies and tools in order to integrate more interactivity in such a type of application.

In this paper, we aim at overcoming these difficulties by proposing a framework that facilitates the design/evaluation of Web-based geographic applications. This framework is composed of three complementary tasks:

1. identifying the desired data that have to be handled by the system;
2. specifying the graphical layout of the application;
3. defining potential end-user interactions.

According to this framework, we have specified a unified model allowing designers to carry out the three tasks mentioned above. In order to assess our proposal, we have integrated this model in a design environment named WIND-Mash. WINDMash proposes visual tools allowing designers to graphically specify the features of the geographic application they want to elaborate. This design task relies on the unified model that we have proposed: each designer’s choice lead to a model part instantiation and is finally translated into source code. Therefore, the operational nature of the unified model allows designers to quickly execute and assess the designed application on any web browser.

The remainder of the paper is structured as follows. In Sect. 2, we present a Web-based geographic application example. Thereafter, in Sect. 3, we describe some related existing systems that allow generating such an example. In Sect. 4, we specify our framework for designing Web-based geographic applications and detail in Sect. 5 its corresponding unified model. Section 6 presents the WIND-Mash design environment. Finally, we conclude in Sect. 7.

2 Motivation: A Use-Case Example

In order to illustrate our approach, we encountered a French school cycling association manager who wanted to organize holidays stages dedicated to its young members coming from elementary school and especially 5th grade classes (9–10 year old children). The manager wanted to share each day into two parts: one dedicated to cycling and one dedicated to school reviews. For this last part, the manager wanted to use references to the famous Tour de France 2012. He decided to rely on a geographic application using a text describing the race and displaying the stages on a map and the corresponding dates on a timeline¹. This application has two main goals. It must first introduce children to the notion of “*département*” and improve their geographic abilities concerning the main towns and “*départements*” visited on the race.

We have used the following French text in our use-case example: “*Pendant l’été 2012, le Tour de France débute en Belgique le 30 juin 2012 et passe par les Vosges et le Jura. Il fait la part belle à la moyenne montagne et aux contre-la-montre, avant l’arrivée à Paris le 22 juillet 2012. Il comporte trois arrivées en montagne et deux contre-la-montre avec une étape de Arc-et-Senans Besançon (38 km) et une autre plus longue de Bonneval à Chartres (52 km)*”. This text is interesting because it mixes spatial and temporal references. Moreover spatial references both refer to cities and to regions. This text is in French because the design environment used to build the geographic application exploits data extraction services able to identify spatial and temporal references inside French texts (see Sect. 6). This text may be translated as follows: “*During summer 2012, the Tour de France sets off from Belgium on 2012, 30th of June, and cross the Vosges and the Jura. It favours medium mountain stages and time-trial stages before finishing in Paris on 2012, 22nd of July. There are three summit finish stages and two time-trial stages with a stage from Arc-et-Senans to Besançon (38 km) and a longer stage from Bonneval to Chartres (52 km)*”.

This application is thus based on textual data and on maps that might be presented into five displayers (Fig. 1):

1. A left-top-side text displayer for the initial text;
2. A main right-part map displayer to both present (highlight) the regions and to zoom-in on the cities’ regions;
3. A centre-top list displayer to highlight the regions either quoted in the text or calculated from the quoted cities;
4. A centre map displayer to zoom-in the cities quoted in the text, and
5. A left-down timeline to display the dates and period quoted in the text.

The application behaviour may be described as follows. When users click on a city in the text, then the system must automatically highlight its corresponding region in the list displayer and in the main map displayer. The system must also zoom-in into this city in the centre map displayer. When users click on a region in the text, this region must be highlighted into the list displayer. When users

¹ <http://erozate.iutbayonne.univ-pau.fr/Nhan/windmash/demo/tourdefrance/>.

Tour de France 2012

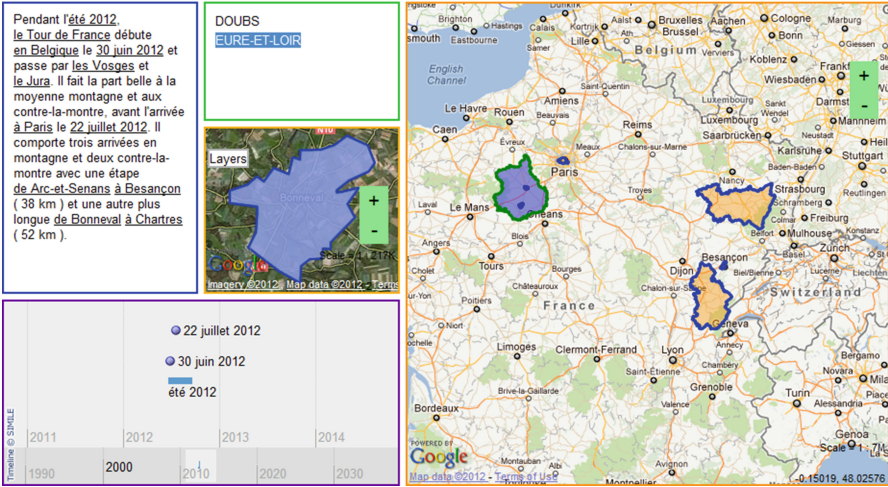


Fig. 1. A Web-based geographic application example presenting the Tour de France 2012

click on a region into the list displayer, this region must be zoomed-in into the main map displayer. When users click on a temporal reference (date or period) in the text, the system must centre on this reference into the timeline displayer.

Developing the interactive Web-based geographic application illustrated in Fig. 1 with the behaviours described above is not a trivial task. It requires:

- Some programming skills, e.g. using JavaScript or AJAX, in order to create an interactive Web application;
- Knowledge about several databases schemas, and especially geographical databases in order to query and get spatial data, such as geolocations on a map;
- Manipulation of Web services (e.g. text indexing services), particularly their inputs/outputs. Naturally, data structures have to be homogenized in order to confront and/or aggregate different services outputs.

Consequently, there is a need of a general framework which contains models and supports some tools for designing Web-based geographic applications.

3 Related Work

End-user mashup programming environments are a new generation of online visual tools enabling users to quickly create, for example, Web-based applications [4]. They rely on metaphors that are easy to grasp by non-professional coders. They may bind together spreadsheets, the flow of linked processing blocks and the visual selection of GUI actions. In [5, 6] several types of mashup environments

have been summarized, some examples are: Yahoo! Pipes², Microsoft Popfly³, Google Mashup Editor⁴, IBM Mashup Center⁵, MashMaker [7], Marmite [8], Vegemite [9], Exhibit [10] and Bill Organiser Portal [11].

However, developing the geographic application example presented in Sect. 2 with these mashup environments is still difficult, while impossible with some of these systems. In fact, many of them do not take into account the specification of end-user interactions and especially the system reactions on the different application components. Furthermore, these systems are not designed to exclusively build geographic applications, hence they do not propose a framework for building such kind of applications.

Some recent works on Web Engineering, especially focusing on geographic applications, proposed architectures based on mapping services. For instance, Mashlight [12] is a Web framework for creating and executing mashup applications by building data processing chains. The generated applications contain by default one mapping component with geolocation information. Besides, Dash-Mash [13] offers to end-users more flexibility for creating geographic applications. Through a set of displayers, they can specify a graphical layout and visualize specific data inside them. Nevertheless, these systems do not allow to design end-user interactions.

SPARQLMotion⁶ is a visual scripting language and an engine for semantic data processing. Scripts implementing sophisticated data services and processing, such as queries, data transformations and mashups, can be quickly assembled with user-friendly graphical tools. However, as the previous systems, it does not provide solutions to specify end-user interactions and a general framework for constructing geographic application.

PhotoMap [14] offers graphical interfaces for navigation and query over some photo collections of users. Especially, with the map-based interface, itineraries followed by users are illustrated and photos with information are linked to specific places. Hence, end-users are able to retrieve where some photos have been taken either by clicking on specific places or by selecting a group of photos. This environment is another use-case for illustrating Web-based geographic applications. Nevertheless, this environment has been hard-coded by an expert and no design framework has been proposed by the authors.

In the next section, we propose an adapted design/evaluation process allowing an author to design and to evaluate by him/herself his/her new Web-based geographic applications. This lightweight and flexible process is based on an adapted model presented in Sect. 5. Our framework called WINDMash allows to easily (visually) instantiate this model in order to automatically generated the corresponding Web-based geographic applications. It should be considered to overcome the limits of the tools presented above.

² <http://pipes.yahoo.com/pipes/>.

³ <http://www.deitel.com/Popfly/>.

⁴ <http://googlemashupeditor.blogspot.com/>.

⁵ <http://www-01.ibm.com/software/info/mashup-center/>.

⁶ <http://www.topquadrant.com/products/SPARQLMotion.html>.

4 Design/Evaluation Process

We designed a process fitted to the development, by novice developers, of Web-based geographic application. The framework depicted in Fig. 2 is composed of three complementary tasks:

1. Identifying the desired data that have to be handled by the geographic application. The data could refer to multimedia contents (e.g., the text written in French in Sect. 2), some extracted information (e.g. in Fig. 1, the list of places automatically identified from the text) and some computed information (e.g. in Fig. 1, the list of “Départements” which corresponds to the list of towns).
2. Specifying the graphical layout of the geographic application. This layout may be composed of several displayers, such as textual displayers, list displayers or map displayers. Thanks to the data that have been defined during the previous step, displayers might display some data sets. For instance, if some data about towns are determined, these towns will be displayed on a specific map viewer.
3. Defining potential end-user interactions on the data that are contained inside displayers. For instance, if a user clicks on a specific place listed in Fig. 1, this place will be highlighted in the text written in French and the map will be focused on this place.

As illustrated in Fig. 2, even if we have ordered the three tasks, it is possible to go backward and forward during the design process. Concretely, at any time the application designer may add, modify and/or remove some data, some displayers or some interactions. Furthermore, it is also possible to design a geographic application without interactions.

As soon as a graphic interface has been elaborated (task 2), designers are able to generate a preview of the application, even if the interface is not completely defined. Obviously, it is possible to update the generated application by repeating our proposed framework.

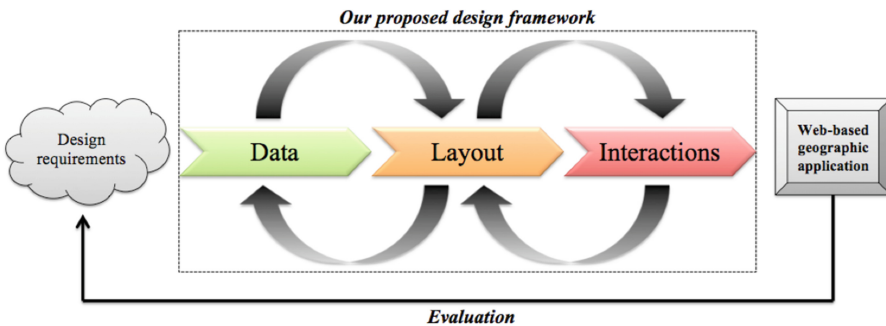


Fig. 2. The Design/Evaluation process [15]

interactions between several representations of a same geographic data (e.g. displaying on a map specific geographic information selected in a text).

As defined in [17], geographic information can be described from a spatial, temporal and/or thematic point of view. Of course, designers may exploit each of these representations in order to display differently the same geographic information (on a text, on a map, on a calendar, etc.). In this context, elaborating the GUI of a geographic application consists in combining several GUI components allowing geographic information to be displayed according to several ways.

Defining interactions in such context deals with specifying what users can do with the geographic information displayed on the screen. As proposed by [18,19], the vocabulary used for designing interactions is based on user action and system reaction: An interaction is defined as a user action triggering a system reaction. A user action corresponds to a selection event (e.g. *click*, *mouse-over*, etc.) on a geographic information in order to select this information or an input event aiming at inputting new geographic information in the system (e.g. defining a marker on a map or writing a set of places in a text component). Each geographic data provided by the user is also considered as an annotation and, according to the way the data is input, the system will define one or more possible representations (e.g. spatial representation if data is input via a map and perhaps a thematic representation if the system can identify the nature of the input data thanks to a geographic database).

System reactions are the system feedbacks resulting from a user action. As suggested in [20], we define two kinds of reactions: An external reaction is characterized by visual modifications of geographic information displayed on the screen. These modifications are carried out by the system and consist in applying visual effects (e.g. *show*, *hide*, *highlight*, *zoom*, etc.) on specific geographic data. An internal reaction is an operation carried out by the system in order to move or copy a data from a GUI component to another, in order to compute new geographic data or to identify a data selected by the user on the screen

6 The WINDMash Environment

WINDMash is a design environment integrating our unified design model. It proposes dedicated tools allowing designers to handle graphically each part of the model in order to create a specific geographic application. Figure 4 (based on the use-case scenario presented in Sect. 2) illustrates these tools:

1. **A pipes editor** which allows to combine different services and to filter the data handled by the application;
2. **A graphical layout editor** which is used to arrange, for instance, mapping components or multimedia contents;
3. **A UML-like sequence diagram builder** which allows to specify potential end-user interactions on the application.

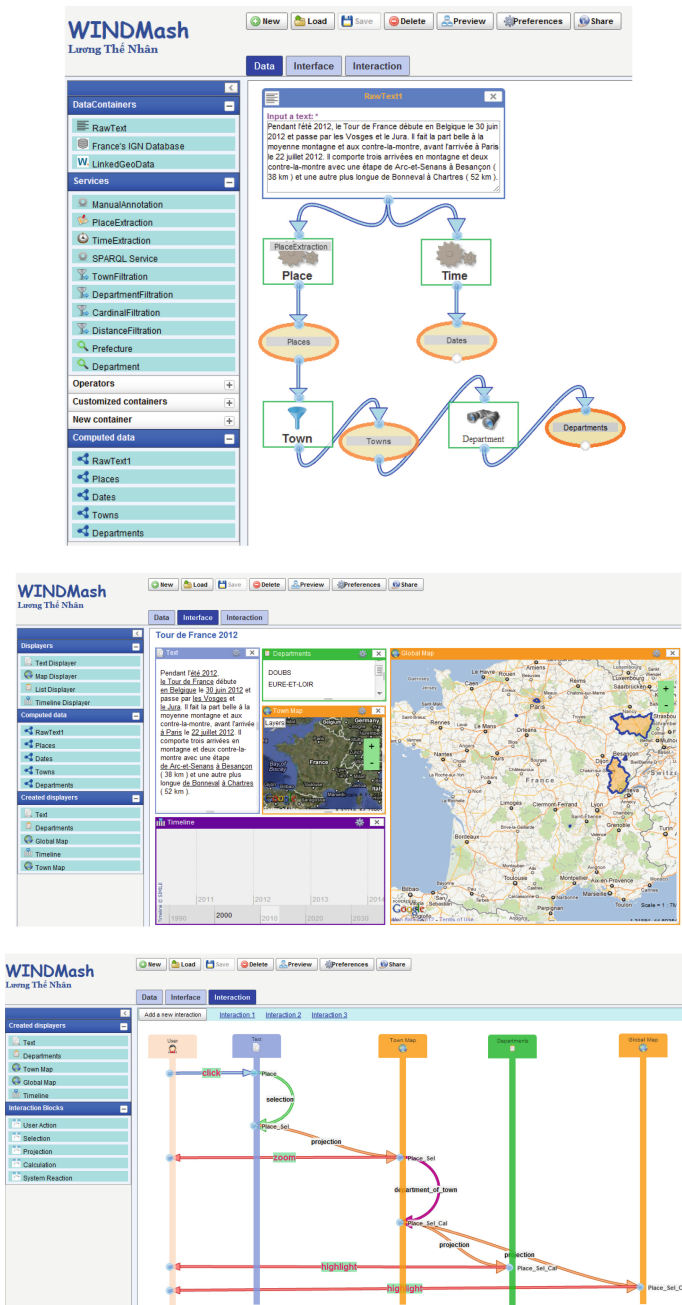


Fig. 4. The WINDMash prototype specifying the example presented in Fig. 1

6.1 A Pipes Editor

In order to manage the data (i.e., contents and annotations) that should be handled by the Web-based geographic application, we have developed a pipes editor allowing designers to create a processing chain containing different services (Fig. 4 on top). While constructing the processing chain, it is possible to visualize at any time the computed data (presented as an oval form) by selecting them with a double click. Each time a dataset is computed, such as the list of extracted places (Places), it generates a RDF/XML description, which corresponds to the data part of the model presented in Sect. 5. These descriptions are also accessible in the menu at the bottom left.

We show in the next section, that these descriptions will be used in the graphical layout editor in order to display the data inside displays.

6.2 A Graphical Layout Editor

The graphical layout editor enables a designer to specify the graphical user interface of his/her Web-based geographic application. Indeed, the designer decides which type of viewer he/she wanted in his/her application (e.g. TextDisplayer, MapDisplayer, ListDisplayer, TimelineDisplayer) and how these displays are organized inside the graphical layout (Fig. 4 in the middle). The menu on the left hand-side indicates the type of displays that may be used by the designer, the available dataset that have been computed with our pipes editor (Sect. 6.1) and the displays that are currently used. In Fig. 4, five displays have been specified: a text displayer, a list displayer, two map displays and a timeline displayer.

Initially, when the designer drags and drops a viewer inside the graphical layout, this viewer is empty, except the map viewer, which contains a map. If the designer wants to display some information inside displays, from the menu, he/she has to drag the computed datasets and to drop them inside a specific viewer. Of course, it is possible to customize each viewer. For instance, the style of a text inside a text viewer may be modified, such as its font, its size, etc. Furthermore, the type of the lists may also be changed, e.g. with or without different kinds of bullets. Finally, different types of maps may be used, such as Google Maps, Yahoo! Maps, IGN Maps. In the next section, we present how it is now possible to specify potential end-user interactions on these components

6.3 A “Sequence Diagram” Builder for Specifying End-User Interactions

In order to specify end-user interactions, we have implemented a UML-like sequence diagram builder which is illustrated in Fig. 4 (bottom). On the left hand-side, the menu is composed of the list of displays that have been specified in Sect. 6.2 and the list of datasets that have been defined in Sect. 6.1. A designer can create several interactions for an application. For example, he/she may create three interactions for the application presented in Sect. 2. The sequence diagram

example illustrated in Fig. 4 describes the following interaction: When a user selects, through a click (see the right arrow with the term *click*), a town contained in the displayer named *Text*; this town is localized (see the right arrow labelled with the term *projection*) in the displayer named *Town Map* and this displayer zooms in on this town (see the left arrow labelled with the term *zoom*). Then, the system computes the department of the selected town (see the curved arrow labelled with the term *department_of_town*); the computed department will be highlighted in the displayer named *Departments* and also in the displayer named *Global Map* (see the left arrow labelled with the term *highlight*).

When the designer has finished building the sequence diagram, the WINDMash prototype handles a global RDF/XML description, which complies with the unified model presented in Sect. 5. Consequently, from this description and our code generator module, the designer could preview the Web-based geographic application by selecting the *Preview* button. If the generated application suits his/her needs, the designer may save the application and/or deploy it on a specific client. Otherwise, the designer may come back to the three WINDMash tools presented in this section in order to add, to modify or to remove some application elements.

7 Conclusion

In this paper, we have presented a framework dedicated to the design of Web-based geographic applications. This framework addresses three complementary tasks which concern the data handled by the application, the graphical layout and the user interactions. We have shown through our unified model for describing such kind of applications that annotations are central in the design process. Indeed, they can be used to describe entities, to display information inside displayers and to specify application behaviours. Furthermore, our proposed framework has been implemented in an online prototype named WINDMash. This prototype contains different visual tools that facilitate the instantiation of our unified model and automatically generates an executable Web-based geographic application.

Currently, our prototype only deals with textual contents. However, the unified model presented in this paper is sufficiently generic to be extended in order to deal with multimedia contents, such as videos, audios and images. Furthermore, the geographic information handled by our WINDMash prototype can be imported from other repositories, for example from the LinkedGeoData⁷ which exploits the spatial information collected by OpenStreetMap⁸. Hence, a future work would consist in developing a mediator between the imported LinkedGeoData datasets and our unified model. Moreover, we plan to extend our model in order to import non-geographic information, such as specific manual annotations.

⁷ <http://linkedgeodata.org>.

⁸ <http://www.openstreetmap.org>.

In the application design process by our prototype WINDMash, the descriptions of three phases (data, layout and interactions) are stored with RDF. Then, they are merged and transformed to HTML + JavaScript codes that can be executed on a Web browser. The originality and the strong point of our approach is to describe the application logic with an RDF format. This choice facilitates the data merging of three phases as well as the code generation. However, to date, WINDMash is not able to detect if a user has described inconsistent behaviour. Our work will be extended to support two points:

- Capacity to verify the consistency of many interactions initialled by a user.
- Capacity to ensure that the generated codes are conformed to a specification designed by a user while they are using the WINDMash visual tools.

References

1. Haklay, M., Singleton, A., Parker, C.: Web mapping 2.0: The neogeography of the GeoWeb. *Geogr. Compass* **2**(6), 2011–2039 (2008)
2. Babar, S.: Accessibility of Web Based GIS Applications: Enhancing Accessibility of Web Based GIS Applications through User Centered Design. LAP LAMBERT Academic Publishing, Verlag (2010)
3. Wilson, D.C., Lipford, H.R., Carroll, E., Karr, P., Najjar, N.: Charting new ground: modeling user behavior in interactive geovisualization. In *Proceedings of the 16th International Conference on Advances in Geographic Information Systems*, pp. 61:1–61:4 (2008)
4. Altinel, M., Brown, P., Cline, S., Kartha, R., Louie, E., Markl, V., Mau, L., Ng, Y.-H., Simmen, D., Singh, A.: Damia: a data mashup fabric for intranet applications. In: *The 33rd International Conference on Very Large Data Bases*, pp. 1370–1373 (2007)
5. Beletski, O.: End user mashup programming environments. Technical report, Telecom Software and Multimedia Laboratory, Helsinki University of Technology (2008)
6. Taivalsaari, A.: Mashware: the future of web applications. Technical report, Sun Microsystems Laboratories (2009)
7. Ennals, R.J., Garofalakis, M.N.: Mashmaker: mashups for the masses. In: *Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data*, pp. 1116–1118
8. Wong, J., Hong, J.I.: Making mashups with Marmite: towards end-user programming for the Web. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI 2007*, pp. 1435–1444. ACM (2007)
9. Lin, J., Wong, J., Nichols, J., Cypher, A., Lau, T. A.: End-user programming of mashups with vegemite. In: *Proceedings of the 14th International Conference on Intelligent User Interfaces, IUI 2009*, pp. 97–106. ACM (2009)
10. Huynh, D.F., Karger, D.R., Miller, R.C.: Exhibit: lightweight structured data publishing. In: *Proceedings of the 16th International Conference on World Wide Web*, pp. 737–746
11. Ro, A., Xia, L.S.-Y., Paik, H.-Y., Chon, C.H.: Bill organiser portal: a case study on end-user composition. In: Hartmann, S., Zhou, X., Kirchberg, M. (eds.) *WISE 2008*. LNCS, vol. 5176, pp. 152–161. Springer, Heidelberg (2008)

12. Albinola, M., Baresi, L., Carcano, M., Guinea, S.: Mashlight: a lightweight mashup framework for everyone. In: *Proceedings of the 2nd Workshop on Mashups, Enterprise Mashups and Lightweight Composition on the Web* (2009)
13. Cappiello, C., Daniel, F., Matera, M., Picozzi, M., Weiss, M.: Enabling end user development through mashups: requirements, abstractions and innovation toolkits. In: *The 3rd International Symposium on End-User Development*, pp. 9–24 (2011)
14. Viana, W., Filho, J.B., Gensel, J., Villanova-Oliver, M., Martin, H.: PhotoMap: from location and time to context-aware photo annotations. *J. Location Based Serv.* **2**, 211–235 (2008)
15. Luong, T. N., Etcheverry, P., Nodenot, T., Marquesuzaà, C., Lopistéguy, P.: WINDMash: a visual mashup environment dedicated to the design of web interactive applications. In: *3rd Workshop on Mash-Up Personal Learning Environments*, Barcelona, Spain, September 2010
16. Etcheverry, P., Laborie, S., Marquesuzaà, C., Nodenot, T., Luong, T.N.: Conception d'applications web géographiques guidée par les contenus et les usages: cadre méthodologique et opérationnalisation avec l'environnement WINDMash. *J. d'Interaction Personne-Système (JIPS)* **3**(1), 1–42 (2014)
17. Gaio, M., Sallaberry, C., Etcheverry, P., Marquesuzaà, C., Lesbegueries, J.: A global process to access documents contents from a geographical point of view. *J. Vis. Lang. Comput.* **19**, 3–23 (2008)
18. Engels, G., Hausmann, J.H., Heckel, R., Sauer, S.: Dynamic meta modeling: a graphical approach to the operational semantics of behavioral diagrams in UML. In: Evans, A., Caskurlu, B., Selic, B. (eds.) *UML 2000*. LNCS, vol. 1939, pp. 323–337. Springer, Heidelberg (2000)
19. Stühmer, R., Anicic, D., Sen, S., Ma, J., Schmidt, K.-U., Stojanovic, N.: Lifting events in RDF from interactions with annotated web pages. In: Bernstein, A., Karger, D.R., Heath, T., Feigenbaum, L., Maynard, D., Motta, E., Thirunarayan, K. (eds.) *ISWC 2009*. LNCS, vol. 5823, pp. 893–908. Springer, Heidelberg (2009)
20. Craig, M.: What is interaction design and what does it mean to information designers? Technical report (1999)

A Context-Aware Recommendation Framework in E-Learning Environment

Phung Do^{1(✉)}, Hung Nguyen¹, Vu Thanh Nguyen^{1(✉)},
and Tran Nam Dung²

¹ University of Information Technology, Vietnam National University
Ho Chi Minh City, Ho Chi Minh City, Vietnam
{phungdtm, hungnp, nguyenvt}@uit.edu.vn

² University of Science, Vietnam National University Ho Chi Minh City,
Ho Chi Minh City, Vietnam
trannamdung@yahoo.com

Abstract. The explosion of world-wide-web has offered people a large number of online courses, e-classes and e-schools. Such e-learning applications contain a wide variety of learning materials which make learners confused to select. In order to address this problem, in this paper we propose a context-aware recommendation framework to suggest a number of suitable learning materials for learners. In the proposed approach, firstly we present a method to determine contextual information implicitly. We then describe a technique to gain ratings from the study results data of learners. Finally, we propose two methods to predict and recommend potential items to active users. The first one is STI-GB for context-aware collaborative filtering (CACF) with contextual modeling approach combined for graph-based clustering technique and matrix factorization (MF). The second one is AVG which predicts ratings based on average calculation method. Experimental results reveal that the proposed consistently outperforms ISMF (combination of Item Splitting and MF), context-aware matrix factorization (CAMF) in terms of prediction accuracy.

Keywords: E-learning · Context · Recommender systems

1 Introduction

E-learning applications play a very important role in online learning support in recent decades. These are network-based applications which provide internet users a flexible environment to learn actively, every time and everywhere with an internet connection. As in the majority of applications in e-commerce, e-learning applications contain the plentiful learning materials namely courses, lessons, references and exercises. As a result, learners face the problem in selecting learning materials which are suitable for their learning levels from the potentially overwhelming number of alternatives.

In order to address this problem, the recommender system (RS) is one of the effective solutions. RSs indicate software tools having techniques to generate suggestions which are suitable items users might prefer [1]. In e-learning, RSs suggest appropriate learning materials (items) for learners (users). Yet RSs do not take into consideration additional information such as time, place, companion and others which

can influence preferences or tastes of users. This additional information is called contextual information or context briefly (more details in Sect. 2). The RSs which deal with contextual information are called context-aware recommender systems (CARSs) [1].

In this paper, we propose a context-aware recommendation framework that uses one of techniques from CARSs to suggest suitable learning materials for learners. It is undeniable that each learner has different competence to study. For instance, an exercise could be easy for a learner, however it might be difficult for others to do. Thus, we are motivated to consider the learning level to make recommendations. We first identify the learning level for each user in the system. This information is determined automatically and considered as context. We then acquire ratings, the rating for a learner (a user) and a question (an item) indicates the probability this learner is able to answer this question correctly. Finally, a method from CARSs and a method from average calculation are applied to recommend learning materials for learners.

In the rest of this paper, the structure is organized as follow: Sect. 2 presents the related work about CARSs and prior work which applies RSs in e-learning environment. We propose a context-aware recommendation framework in Sect. 3. Section 4 is experimental setup to compare my proposal with some algorithms. Finally, Sect. 5 is conclusion and future works.

2 Related Works

Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application [2]. Entity is often a user, an item and the rating from a user over an item in terms of RSs.

There are three ways [3] to obtain contextual information include:

- *Explicitly* from the relevant objects by asking direct questions or eliciting through other means.
- *Implicitly* from the data or the environment such as the change in location automatically detected by devices or the time stamp of a transaction.
- *Inferring* the context using statistical or data mining methods.

CARSs indicates RSs which incorporate contextual information into recommendation process to model and predicting tastes of users. As in RSs, collaborative filtering (CF) [1, 4] is often used in CARSs to make recommendations. Recall that CF uses a 2-dimensions matrix (*the rating matrix* $U \times I$) made up by a list of users (U) and a list of items (I). The rating matrix represents preferences' users which are explicit ratings with scale 1–5 or implicit indications such purchasing frequencies or click-throughs [4]. Because of the appearance of context, the rating matrix is extended as a multi-dimensional matrix (denoted $U \times I \times C$) with contexts. There are missing values in the matrix where users did not give their preferences for certain items (in certain contexts) and CF has to predict them. CF is extended to deal with contextual information called context-aware collaborative filtering (CACF).

The prior work of CARSs are implemented in e-commerce [5], entertainment (music [6], tourism [7], movie [3, 8]) and food [9] domains).

Regarding e-learning domain, recommendation task is implemented with methods in RSs without the using of context. In particular, in [10], neighborhood-based CF is applied to recommend suitable learning materials. Additionally, [11] maps educational data into $U \times I$ matrix [4] and applies matrix factorization technique to predict student performances. Furthermore, recommendation processes use hybrid methods as well. Specially, [12] applies both user-based and rule-based methods to suggest relevant courses for learners. A scientific paper recommendation engine using model-based CF and hybrid techniques is built in [13]. The work in [14] proposes equations to adapt CF into e-learning meanwhile [15] combines RS with ubiquitous computing to enhance learning with memory-based CF and association mining techniques in botanic subject. In [16], a recommender system with AprioriAll and memory-based CF methods is constructed to suggest java lessons. E-learning RSs also use item's attributes to generate suggestions. The work in [17, 18] propose hybrid attribute-based methods with CF and content-based techniques to recommend books for learners, attributes consist of subjects, education levels, prices and authors or implicit attributes from history ratings of learners. The work in [23] proposes an approach based on community detection which divides users into several groups with similar interest, study ability and others contextual information and then provides appropriate recommendations made based on students belonging to their group. The work in [24] develops a trust-aware collaborative filtering scheme based on learning styles, knowledge levels and trustworthiness of learners in recommendation process to ensure that recommended resources are suggested by trustworthy learners. The work in [26] proposes a framework that employs the k-means clustering technique to identify groups of similar students and tasks based on the corresponding skill profiles then imposes the locality preserving constraints into the weighted regularized nonnegative matrix factorization for predicting student performance.

In relation to context-aware e-learning systems, [19] analyzes the user's knowledge gap (context) to filter suitable learning contents instead of using a recommendation technique. [25] proposes a graph-based framework to model and incorporate contextual information into the recommendation process. Prior works mentioned above do not use any CACF technique. Therefore, the main contribution of this paper is to build a context-aware recommendation framework in e-learning environment with a CACF technique and average calculation method to recommend learning materials for learners.

3 Proposed Context-Aware Recommendation Framework

The proposed context-aware recommendation framework first of all infers context and collects ratings extracted from study results data and questions data. A rating prediction method is applied with the rating matrix obtained from the step above to build recommender model. The model is retrieved to predict ratings and to produce recommendations. We propose two rating prediction methods are Average calculation method and Context-aware collaborative filtering (CACF) algorithm with contextual modeling approach combined from a clustering technique and matrix factorization method.

First Through user’s interfaces, learners take *examinations* or do *practice tests* and study results data is then recorded. The study results data contains information about the timestamp, answers, score, correct answers, incorrect answers, related-lessons, users and others (Fig. 1).

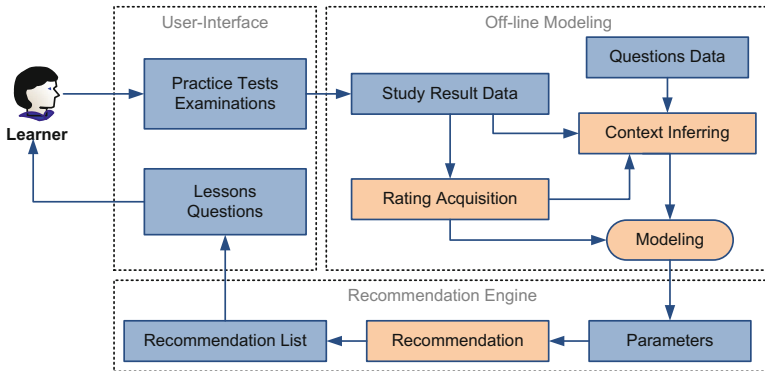


Fig. 1. The recommendation process

Second At server side with offline mode, the *study results data* is regularly processed with three main steps executed consecutively.

- *Context Inferring:* The *study results data* and *questions data* of system are used to infer context. As a result, the *C* component of the rating matrix is identified.
- *Rating Acquisition:* The main purpose of this step is to collect ratings extracted from the study results data. This step aims to find out the *U*, *I* and ratings components of the rating matrix.
- *Modeling:* An algorithm is applied with the rating matrix obtained from two steps above to build the recommender model which is stored/updated in the system database.

Finally *Recommendation* is processed. The model is retrieved to predict ratings and to produce recommendations. The *recommendation list* is normally a list of *lessons* or a list of *questions* displayed in the user’s interface. The users continue taking examinations and doing practice tests to enrich the study results data for the next modeling processes.

3.1 Context Inferring

Context in system is the learning level which depicts the academic performance of a learner in the fixed period of time. We define the learning level (context) with 7 values: beginner, elementary, pre-intermediate, intermediate, upper-intermediate, advanced and fluent with the corresponding indexes {0, 1, 2, 3, 4, 5, 6}. The learning level of a new

learner in the system is marked as 0 and it is then modified when this learner does practice tests or examinations.

We propose a method to determine automatically the learning level for each learner in the system. To infer context, the hard level attribute of questions and the study results data are examined. The hard level of a question is manually pre-defined by teachers and it has value range $[1, h]$, $h = 3$ in our application. For each the u^{th} learner, a score vector when the context inferring step is occurred, $s_u = (z_1, z_2, \dots, z_j, \dots, z_h), j = 1 \dots h$, is constructed. The value of z_j is computed as the number of times the u^{th} learner answers correctly questions with the hard level j , divided by the total number of times this learner answers these questions. Note that $z_j = -1$ when the u^{th} learner does not answer any question with the hard level j . Then the value of $Score_u$ is calculated as follow:

$$Score_u = \frac{\sum_{j=1, z_j \neq -1}^h jz_j}{\sum_{j=1, z_j \neq -1}^h j} \tag{1}$$

The value of $Score_u$ indicates the study result of the u^{th} user from doing exercises until the context inferring step is occurred. Having the value of $Score_u$, we use Table 1 to decide the context of the u^{th} user. This table is based on the marking scale (0–10) and ranking in our university, we transform it to scale (0–1) to identify the learning level.

Table 1. The $Score_u$ value table for context inferring

Range of $Score_u$	Context value
[0, 0.3)	0
[0.3, 0.5)	1
[0.5, 0.6)	2
[0.6, 0.7)	3
[0.7, 0.8)	4
[0.8, 0.9)	5
[0.9, 1]	6

3.2 Ratings Acquisition

As in the vast majority of RSs, the rating 1–5 or like-dislike scale is the most commonly used and it often shows the preference of a user to an item (in a certain context). However, giving the rating for each question (item) in a long practice test or an examination seems to be unreasonable. Moreover, a question learners might like could be very easy and vice versa. Thus, to obtain the ratings, we propose the following expression:

$$r_{uic} = \frac{p_{uic}}{s_{uic}} \quad (2)$$

Where r_{uic} is the probability the u^{th} learner answers correctly the i^{th} question in context c , s_{uic} indicates the total number of times the u^{th} learner answers the i^{th} question in context c and p_{uic} is the number of times this user gives the correct answers in context c .

3.3 Modeling

A rating prediction method is applied to predict ratings for unrated items. The rating matrix gained is used to build the recommender model. The recommender model is periodically re-built and stored/updated in a database. Modeling is an off-line mode process; therefore, it does not affect the speed of real-time response to the user's interface. We propose two ratings method to gain the rating matrix are *Rating prediction based on graph-based clustering technique* and *Rating prediction based on average calculation method*.

Rating Prediction Based on Graph-Based Clustering Technique (STI-GB). The algorithm is basically based on clustering technique and matrix factorization in RSs. This algorithm with main idea "similar pairs (*item, context*) are clustered and converted to new products to reduce contextual dimensions. After that, matrix factorization is applied combined with analyzing the effect of contextual information to predict and recommend potential items to the active users" consists of 5 main steps.

Step 1. Building item profiles

Each $ItemProfile(i, c)$ in the item profiles is a representation vector for the i^{th} item and context c , such vector contains ratings for all users pertaining to this item and context. $ItemProfile(i, c) = (r_{1ic}, r_{2ic}, \dots, r_{uic}, \dots, r_{nic}), u = 1 \dots n$, r_{uic} is the rating from the u^{th} user for the i^{th} item in context c and $r_{uic} = -1$ if there is no rating. Notice that, c is the combination value of contexts. For instance, we have two contexts X and Y with numbers of values in X and Y are 2 and 3 respectively. Therefore, we have $2 \times 3 = 6$ combinations of contextual information and with 20 items, there are $6 \times 20 = 120$ item profiles.

Step 2. Clustering

The main purpose of this step is to identify similar trends in giving the preferences or ratings to items in different contexts. It is handled by applying a clustering technique to obtain k clusters from item profiles (*step 1*).

There is a number of clustering methods such as k-Means, G-Means, DBScan, Clara and others using Manhattan, Euclidean and other distances to find clusters. However, neighborhoods in RSs are frequently identified through similarity values between users or items, which normally calculated by ratings of co-rated items. Therefore, we present the graph-based clustering technique adapting collaborative filtering.

Item Profiles set (D) contains N numeric vectors. The D set is considered as an undirected graph $D = \{V, E\}$, $V = \{V_1, V_2, \dots, V_N\}$ is a set of N vertices or nodes corresponding to N numeric vectors and E is a set of edges or lines between to vertices. The weight value between $V_i = (v_{i1}, v_{i2}, \dots, v_{in})$ and $V_j = (v_{j1}, v_{j2}, \dots, v_{jn})$, denoted $weight(V_i, V_j)$ is cosine value computed by following formula:

$$weight(V_i, V_j) = \frac{\sum_{k=1}^n v_{ik}v_{jk}}{\sqrt{\sum_{k=1}^n v_{ik}^2} \sqrt{\sum_{k=1}^n v_{jk}^2}}, i \neq j \tag{3}$$

Where n is the dimension of vectors V_i and V_j (or the number of users). It is noticeable that $weight(V_i, V_j)$ is computed with values v_{ik} and v_{jk} , $k = 1 \dots n$ larger than -1 from vector V_i and V_j respectively. The value of $weight(V_i, V_j)$ indicates the similarity between item profiles V_i and V_j which means that users intend to give the similar ratings for the item profiles V_i and V_j .

For each vertex V_i , $i = 1 \dots N$, let $w_j = weight(V_i, V_j)$, $j \neq i$. We build an edge from vertex V_i to vertex V_k if there is only one $j = k$ and $max\{w_j\} = w_k$. In case there is more than one vertex V_k and $max\{w_j\} = w_k$, we will choose the vertex V_k with lowest index k .

In order to cluster data input D into k clusters, one of the graph methods such as Warshall, DFS (Depth First Search) or BFS (Breadth First Search) is applied to find out connected components of undirected graph D . As the result of this stage, the k connected components are obtained and it also means k clusters. In particular, BFS algorithm is used to find k clusters in this paper.

Step 3. Building 2-dimensional matrix

This step aims to reduce contextual information dimension so that the sparsity problem [20] can be partially tackled. After clustering (*step 2*), a new products set $P = \{p_1, p_2, \dots, p_j, \dots, p_k\}$ including k new products corresponding to k clusters is constructed. The rating value for a new product p_j for each user is the average rating of this user in the j^{th} cluster (notice that we only consider rating values larger than -1 to compute average rating for each user). The matrix $U \times I \times C$ (*User* \times *Item* \times *Context*) is transformed into the 2-dimensional matrix $U \times P$ (*User* \times *Product*) in this stage. Regarding testing phase, the predictive rating value for the i^{th} item in context c is replaced by the predicting value for a new product p_j with the j^{th} cluster contains $ItemProfile(i, c)$. Figure 2 illustrates an example of STI-GB algorithm from step 1 to step 3.

Step 4. Analyzing the effect of contextual information

The value of $ContextEffect(c, u)$ indicates the effect value from context c to the u^{th} user. The value of $ContextEffect(c, u)$ is computed as the average rating value of the u^{th} user in context c (symbolized $avg(u, c)$) subtracts the overall average rating value of the u^{th} user (symbolized $avg(u)$).

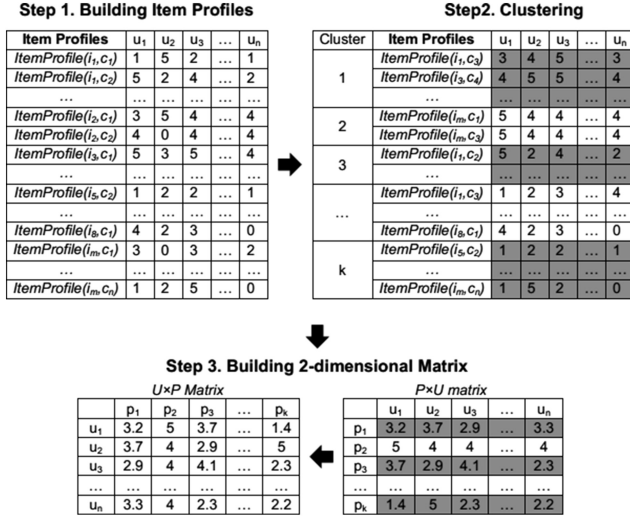


Fig. 2. An example of STI-GB algorithm from step 1 to step 3

$$ContextEffect(c, u) = avg(u, c) - avg(u) \tag{4}$$

Step 5. Predicting

In order to predict the rating value for the u^{th} user, the i^{th} item in context c , applying first matrix factorization (SVD) [1] technique with 2-dimensional matrix $U \times P$ from step 3 with result $MF(u, p_j)$ depicts the predictive rating value for the u^{th} user and the j^{th} product p_j (the j^{th} cluster contains $ItemProfile(i, c)$).

Then, the predictive rating value for the u^{th} user, the i^{th} item in context c is computed as following expression:

$$\hat{r}_{uic} = MF(u, p_j) + ContextEffect(c, u) \tag{5}$$

The predictive rating values are adjusted in range rating scale $[min, max]$. The value of \hat{r}_{uic} is $min(max)$ when $MF(u, p_j) + ContextEffect(c, u)$ smaller (larger) than $min(max)$.

Rating Prediction Based on Average Calculation Method (AVG) User u 's rating of item i in c contextual, which denoted by \hat{r}_{uic} is estimated to be approximately user u rating average in c contextual.

$$\hat{r}_{uic} = avg(u, c) \tag{6}$$

If user u don't have any rating in c contextual, user u 's rating of item i is estimated to be approximately user u 's rating average.

$$\hat{r}_{uic} = avg(u) \quad (7)$$

New user u 's rating of item i in c contextual is estimated to be approximately all users' rating average of item i in c contextual.

$$\hat{r}_{uic} = avg(i, c) \quad (8)$$

New user u 's rating of new item i in c contextual is estimated to be approximately all users' rating average of all items in c contextual

$$\hat{r}_{uic} = avg(c) \quad (9)$$

In the remaining cases, user u 's rating of item i is estimated to be approximately all users' rating average of all items in all contexts.

3.4 Recommendation

As usual in RSs (CARSs), an algorithm is applied to predict ratings for unrated items and then items with high predicted ratings are recommended for the active user. However, in learning environment, there are some questions/lessons that learners need to do/study several times to help learners consolidate their knowledge. Therefore, recommendation in learning environment have to focus on learning resources, we proposed two recommendation tasks include: (1) Lessons recommendation and (2) questions recommendation.

Lessons Recommendation. It is true to say that if a lesson is extremely hard to understand and the probability a learner gives the answer for a question is high. Thus, this task is to make learners pay attention to important lessons which they need to understand clearly. Lessons are recommended immediately when a learner finishes an examination and it offers a list of lessons which this learner should review.

Given $L = \{l_0, l_1, \dots, l_j, \dots, l_k\}$, $j = 0..k$ contains lessons whose questions appear in the examination. For each lesson l_j , find Q_j set, which includes questions belong to this lesson and predict ratings for questions in Q_j . The predicted rating for l_j is the average rating of questions in Q_j . Then such ratings are ranked in ascending order. Lessons with ratings smaller than a threshold τ_q are recommended to the active learner.

The rating for a lesson in this case reveals the probability a learner can answers correctly all questions belong to this lesson, that is the reason why the lessons with the smallest ratings would be on top in the recommendation list and the learner should make them priorities to review.

Questions Recommendation. Unlike RSs (CARSs) normally recommend items which users do not give their assessments, we consider all questions to recommend. This is because, an extremely difficult question could be recommended many times although there are ratings for this question.

Although we consider all questions include questions done by learners, we just recommend done questions if their predicted rating values are smaller than a threshold τ_q . For instance, a learner answers correctly an easy question 3 times, and the predicted value is approximately 0.82 which is larger than the threshold $\tau_q = 0.6$, then we do not recommend this question.

We propose two options for questions recommendation:

1. The first one is overall practice test which has the same structure as a real examination with questions related many lessons included. Learner is required to provide the number of questions.
2. The second one is the personalized practice test which learners can choose lessons they want to practice and how many questions belongs to the each chosen lesson.

Learners can provide duration and the hard level to obtain the suitable practice test. There are two types of hard level: ascending (default) and descending. The system predicts ratings for all questions related to the learner's configuration and recommends questions with highest predicted ratings in case the ascending hard level is selected and vice versa.

4 Experiments and Results

We compare the AVG with Item Splitting [21], Context-Aware Matrix Factorization (CAMF) [22] and STI-GB methods. In particular, the Item Splitting method which reduces contextual dimensions and applies matrix factorization to predict unknown ratings is called ISMF. ISMF, CAMF and STI-GB algorithms are based on Matrix Factorization technique (SVD) [1] and these algorithms are configured same parameters.

We use five real-world datasets for comparisons and evaluations. The datasets are used to evaluate the efficiency of algorithms in CARS in general. Additionally, three lasts datasets related to learning are used to examine the suitability of algorithms in e-learning environment. The first one is AIST context-aware food preference dataset called Food in [9], this dataset after pre-processing duplicated records contains 5,300 ratings from 212 real users over 20 food menus in the real and imaginary level of hungry situations. Therefore, we use these situations along with the levels of hungry as contextual information. The second one is LDOS-Comoda dataset called Comoda which is used in [8]. Comoda includes 2,248 ratings from 82 users in 1,225 movies, rating values are judged along with 12 different contextual information. The third one is Spoj collected from the online judge system www.spoj.com. This dataset after pre-processing includes 899,694 submissions with the results are *true* or *false* from 16,783 registered users in 2,968 public problems along with learning level as contextual information. Learning level is deduced based on users' submission process, after each 10 problems that users solved, learning level is changed. The fourth one is UIT_cqui. This is the learning dataset of the University of Information Technology's regular program from 2006 to 2013. The dataset contains 6,145 students, 342 subjects and 214,905 scores range from 0 to 10. The status of courses and level of students are contexts. Level contextual of students is deduced based on students' learning process,

after each semester that students completed, level contextual is changed. The last one is UIT_txqm. This is the learning dataset of the University of Information Technology’s distance learning program. The dataset contains 22,015 students, 92 subjects and 971,886 scores range from 0 to 10. The dataset’s contexts is the same as UIT_cqui dataset’s contexts (Table 2).

Table 2. The parameters of datasets

Dataset	User number	Product number	Rating number	Context number	Rating value	Sparsity
Food	212	20	5,300	2	{1,2,3,4,5}	79.17 %
Comoda	82	1225	2,248	12	{1,2,3,4,5}	99.99 %
Spoj	16,783	2,968	899,694	1	{0,1}	99.74 %
UIT_cqui	6,145	342	214,905	2	[0,10]	99.76 %
UIT_txqm	22,015	92	971,886	2	[0,10]	98.86 %

Datasets are partitioned into 5 disjoint sets and we apply 5-fold cross validation with the most common evaluation metrics such as Mean Absolute Error (*MAE*) and Root Mean Square Error (*RMSE*) [1] to estimate algorithms (Table 3).

Table 3. The result of evaluation

Dataset	Metric	ISMF	CAMF	STI-GB	AVG
Food	MAE	0.876	0.843	0.853	0.693
	RMSE	1.104	1.065	1.118	0.966
Comoda	MAE	0.841	0.884	0.809	0.827
	RMSE	1.067	1.172	1.055	1.037
Spoj	MAE	0.155	0.155	0.157	0.230
	RMSE	0.394	0.382	0.395	0.342
UIT_cqui	MAE	1.443	1.278	2.159	1.356
	RMSE	2.092	1.878	2.553	1.819
UIT_txqm	MAE	1.402	1.288	1.599	1.032
	RMSE	2.101	1.957	2.137	1.400

Results of using rating prediction based on average calculation method is better in most cases. With Food and UIT_txqm datasets, compare to ISMF, CAMF, STI-GB algorithms, average calculation method decreases MAE 20.89 %, 17.79 % and 18.75 %, RMSE 12.5 %, 9.3 % and 13.06 % in Food dataset, decreases MAE 26.39 %, 19.88 % and 35.46 %, RMSE 33.37 %, 28.46 % and 34.49 % in UIT_txqm dataset. With Comoda, Spoj and UIT_cqui datasets, RMSE results are lowest when using average calculation method. Compare to ISMF, CAMF and STI-GB, this method decreases RMSE 2.81 %, 11.52 % and 1.71 % in Comoda dataset, 13.2 %, 10.47 % and 13.42 % in Spoj dataset, 5.2 %, 4 % and 5.3 % in UIT_cqui. Using average calculation method, MAE in Comoda dataset is greater than STI-GB 2.23 % but lower than

ISMF, CAMF 1.67 % and 6.45 %, MAE in UIT_cqui is greater than CAMF 6.1 % but lower than ISMF, STI-GB 6.03 % and 37.19 %.

Furthermore, this method has much lower cost implementation than the implementations of using algorithms based on matrix factorization ISMF, CAMF, STI-GB.

5 Conclusion and Future Works

In this paper, we propose a context-aware recommendation framework which recommends learning materials for learners. We first present the method to determine contextual information implicitly. We then describe the technique to gain ratings from the study results data of learners. Finally, we propose two methods to predict and recommend potential items to the active users. The first one is STI-GB for CACF with contextual modeling approach. This method clusters similar pairs (*item*, *context*) to convert the multi-dimensional matrix $User \times Item \times Context$ into the 2-dimensional matrix $User \times Product$ to reduce sparsity problem. The second one is AVG which predicts ratings based on average calculation method. We also compare the STI-GB and the AVG with different clustering techniques on particular datasets.

In the future, we intend to incorporate the forecasting techniques into CARS to improve the quality of recommendations.

Acknowledgments. This research is the output of the project “Context-aware Recommender System and applying for E-Learning Recommendation” under grant number D2013-04 which belongs to University of Information Technology - Vietnam National University Ho Chi Minh City.

References

1. Ricci, F., Rokach, L., Shapira, B., Kantor, P.B.: Recommender Systems Handbook, 1st edn, pp. 217–249. Springer, New York (2010)
2. Dey, A.K.: Understanding and using context. *Pers. Ubiquit. Comput.* **5**(1), 4–7 (2001)
3. Adomavicius, G., Tuzhilin, A.: Context-aware recommender systems. In: Proceedings of the 2008 ACM Conference on Recommender Systems, RecSys 2008, pp 335–336. ACM, New York, NY, USA (2008)
4. Su, X., Khoshgoftaar, T.M.: Collaborative filtering for multi-class data using belief nets algorithms. In: Proceedings of the 18th IEEE International Conference on Tools with Artificial Intelligence, ICTAI 2006, pp. 497–504. IEEE Computer Society, Washington, DC, USA (2006)
5. Panniello, U., Gorgoglione, M.: Incorporating context into recommender systems: an empirical comparison of context-based approaches. *Electron. Commer. Res.* **12**(1), 1–30 (2012)
6. Baltrunas, L., Kaminskas, M., Ludwig, B., Moling, O., Ricci, F., Aydin, A., Lücke, K.-H., Schwaiger, R.: InCarMusic: context-aware music recommendations in a car. In: Huemer, C., Setzer, T. (eds.) EC-Web 2011. LNBI, vol. 85, pp. 89–100. Springer, Heidelberg (2011)
7. Baltrunas, L., Ludwig, B., Peer, S., Ricci, F.: Context relevance assessment and exploitation in mobile recommender systems. *Pers. Ubiquit. Comput.* **16**(5), 507–526 (2012)

8. Odic, A., Tkalcic, M., Tasic, J.F., Kosir, A.: Relevant context in a movie recommender system: users opinion vs. statistical detection. In: Adomavicius, G. (ed.) *Proceedings of the 4th Workshop on Context-Aware Recommender Systems in Conjunction with the 6th ACM Conference on Recommender Systems (RecSys 2012)*, vol. 889 (2012)
9. Ono, C., Takishima, Y., Motomura, Y., Asoh, H.: Context-aware preference model based on a study of difference between real and supposed situation data. In: Houben, G.-J., McCalla, G., Pianesi, F., Zancanaro, M. (eds.) *UMAP 2009. LNCS*, vol. 5535, pp. 102–113. Springer, Heidelberg (2009)
10. Soonthornphisaj, N., Rojsattarat, E., Yim-ngam, S.: Smart e-learning using recommender system. In: Huang, D.-S., Li, K., Irwin, G.W. (eds.) *ICIC 2006. LNCS (LNAI)*, vol. 4114, pp. 518–523. Springer, Heidelberg (2006)
11. Thai-Nghe, N., Drumond, L., Krohn-Grimberghe, A., Schmidt-Thieme, L.: Recommender system for predicting student performance. *Procedia Comput. Sci.* **1**(2), 2811–2819 (2010)
12. Tan, H., Guo, J., Li, Y.: E-learning recommendation system. In: *Proceedings of the 2008 International Conference on Computer Science and Software Engineering, CSSE 2008*, vol. 05, pp. 430–433. IEEE Computer Society, Washington, DC, USA (2008)
13. Tang, T., McCalla, G.I.: Evaluating a smart recommender for an evolving e-learning system: a simulation-based study. In: Tawfik, A.Y., Goodwin, S.D. (eds.) *Canadian AI 2004. LNCS (LNAI)*, vol. 3060, pp. 439–443. Springer, Heidelberg (2004)
14. Bobadilla, J., Serradilla, F., Hernando, A.: Collaborative filtering adapted to recommender systems of e-learning. *Knowl. -Based Syst.* **22**(4), 261–265 (2009)
15. Wang, S.L., Wu, C.Y.: Application of context-aware and personalized recommendation to implement an adaptive ubiquitous learning system. *Expert Syst. Appl.* **38**(9), 10831–10838 (2011)
16. Klasnja-Milicevic, A., Vesin, B., Ivanovic, M., Budimac, Z.: E-learning personalization based on hybrid recommendation strategy and learning style identification. *Comput. Educ.* **56**(3), 885–899 (2011)
17. Salehi, M., Kmalabadi, I.N.: A hybrid attributebased recommender system for e-learning material recommendation. *IERI Procedia* **2**, 565–570 (2012)
18. Salehi, M., Kmalabadi, I.N., Ghoushchi, M.B.G.: A new recommendation approach based on implicit attributes of learning material. *IERI Procedia* **2**, 571–576 (2012)
19. Schmidt, A., Winterhalter, C.: User context aware delivery of e-learning material: approach and architecture. *J. Univ. Comput. Sci. (JUCS)* **10**, 28–36 (2004)
20. Su, X., Khoshgoftaar, T.M.: A survey of collaborative filtering techniques. In: *Advances in Artificial Intelligence*, vol. 2009 (2009)
21. Baltrunas, L., Ricci, F.: Context-based splitting of item ratings in collaborative filtering. In: *Proceedings of the Third ACM Conference on Recommender Systems*, pp. 245–248. ACM, New York, NY, USA (2009)
22. Baltrunas, L., Ludwig, B., Ricci, F.: Matrix factorization techniques for context aware recommendation. In: *Proceedings of the Fifth ACM Conference on Recommender Systems, RecSys 2011*. ACM, New York, NY, USA 301–304 (2011)
23. Senthil, K.V., Sankar, A., Kiruthikaa, K.: Community based recommendation in elearning systems. *J. e-Learn. Knowl. Soc.* **10**(1), 51–61 (2014)
24. Dwivedi, P., Bharadwaj, K.K.: Effective trust-aware e-learning recommender system based on learning styles and knowledge levels. *J. Educ. Technol. Soc.* **16**(4), 201–216 (2013)
25. Wu H., Yue K., Liu X., Pei Y.J., Li B.: Context-aware recommendation via graph-based contextual modeling and postfiltering. *Int. J. Distrib. Sens. Netw.* **2015** (2015)
26. Hwang, C.S., Su, Y.C.: Unified clustering locality preserving matrix factorization for student performance prediction. *IAENG Int. J. Comput. Sci.* **42**(3), 86–94 (2015)

Automatic Evaluation of the Computing Domain Ontology

Chien D.C. Ta^(✉) and Tuoi Phan Thi

Faculty of Computer Science and Engineering,
HCMC University of Technology, Ho Chi Minh City, Vietnam
{chientdc, tuoi}@cse.hcmut.edu.vn

Abstract. Ontology plays an important role in the recent years. Its applications now are more popular and variety. Ontologies are used in the different areas related to Information Technology, Biology, and Medicine, especially in Information Retrieval, Information Extraction, and Question Answering. Ontologies capture background knowledge by providing relevant terms and the formal relations between them, so that they can be used in a machine-processable way. Depending on the different applications, the structure of ontologies has been built and designed with different models. Good ontologies lead directly to a higher degree of reuse and a better cooperation over the boundaries of applications and domains. However, there are a number of challenges that must be faced when evaluating ontologies. In this paper, we propose a novel approach based on data-driven and information extraction system for evaluating the lexicon/vocabulary and consistency of a domain specific ontology. Furthermore, we evaluate the ontological structure and the relations of some terms of the ontology.

Keywords: Ontological evaluation · Domain specific ontology · Information extraction

1 Introduction

The methodological approaches for evaluating ontologies have become an active field of research in recent years. Depending on different applications and the structure of ontologies, ontology evaluation can be applied in many ways. There are a lot of researches which are relevant to this field. Pérez [1] proposed a method to evaluate ontology which consists of two steps. The first one is to describe a set of initial and general ideas that guide the evaluation of ontologies. The second one is to apply empirically some of these ideas in the evaluation of the Bibliographic-Data ontology. Fahad et al. [2] proposed a framework for evaluating ontology. His proposal presented the ontological errors based on design principles for evaluation of ontologies. It provided the overview of ontological errors and design anomalies that reduces reasoning power and creates ambiguity while inferring errors from concepts. Velardi [3] proposed an approach for evaluation of an actual ontology-learning system. His approach consists of twofold: first, he provided a detailed quantitative analysis of the ontology learning algorithms. Second, he automatically generated natural language descriptions

of formal concept specifications in order to facilitate per concept qualitative analysis by domain specialists. In general, there are a number of researches related to evaluating ontology. However, these researches evaluate the ontologies based on either the design principles or the vocabulary of ontology. They do not combine all of them in order to evaluate ontology more accurately. In this paper, we introduce an approach for ontological evaluation based on the structure, axioms of the ontology and semantics among concepts.

Our key contributions are as follows: (i) the lexicon/vocabulary and consistency of a domain specific ontology are evaluated, which based on the data-driven related to computing domain; (ii) we build an information extraction system to evaluate the lexicon/vocabulary and consistency of the ontology; (iii) we also evaluate the ontological structure and the relations of terms based on data constraints.

The rest of this paper is organized as follows: Sect. 2 examines related work and overviews a sample of approaches; Sect. 3 introduces the proposed methodology; Sect. 4 illustrates the experimental results; Sect. 5 discusses the conclusions and future works.

2 Related Work

There are a number of frameworks for ontology evaluation, such as OntoClean [4], OntoManager [5], OntoMetric [6], etc. Each framework has its own advantages and disadvantages depending on the complexities of ontologies. As outline from Netzer et al. [7] proposed a new method to evaluate a search ontology, which relied on mapping ontology instances to textual documents. On the basis of this mapping, he evaluated the adequacy of ontology relations by measuring their classification potential over the textual documents. This data-driven method provided concrete feedback to ontology maintainers and a quantitative estimation of the functional adequacy of the ontology relations towards search experience improvement. He specifically evaluated whether an ontology relation can help a semantic search engine support exploratory search. Soysal et al. [8] built the domain specific ontology focusing on movie domain. He used three measures, namely Precision, Recall, and F-measure for ontology evaluation.

The above-mentioned researches and frameworks lack in a general approach for evaluating the ontologies by combining the features of the ontologies, such as, the structure of the ontologies, the concepts and axioms in the ontologies. According to Obrst et al. [9], he suggests that there are a number of methods to evaluate the ontologies. They are:

- The evaluation of the use of an ontology in an application.
- The comparison against a source of domain data.
- Assessment by humans against a set of criteria.
- Natural language evaluation techniques. Natural language processing tasks such as information extraction, question answering and abstracting are knowledge-hungry tasks. It is, therefore, natural to consider evaluation of ontologies in terms of their impact on these tasks.

- Using reality as a benchmark. Here the notion of a “portion of reality” is introduced, to which the ontology elements are compared.

In order to evaluate a more effective one for the ontology, in this paper, we will propose an approach combining the data-driven and data constraints related to computing domain. Moreover, we build an information extraction system based on this ontology for evaluating the accuracy of concepts in the ontology.

3 Automatic Evaluation of the Computing Domain Ontology

3.1 Overview of the Computing Domain Ontology

Ontology is a formal and explicit specification of a shared conceptualization of a domain of interest. Their classes, relationships, constraints and axioms define a common vocabulary to share knowledge. Conceptualization refers to an abstract model of some phenomenon in the world. Explicit specification means that the type of concepts used and the limitations of their use are explicitly defined. Formal specification refers to the fact that the ontology should be machine-readable. Shared knowledge reflects the notion that ontology captures consensual knowledge, which is not private to some individual but accepted by a group.

Formally, an ontology can be defined as the tuple [10]:

$$O = (C, I, S, N, H, Y, B, R)$$

Where,

C , is set to consist of classes. In this ontology, C represents categories of computing domain (for example, “Artificial Intelligent, hardware devices, NLP” $\in C$)

I is set of instances belong to categories. In this ontology, set I consists of computing vocabulary (for example, “robotic, Random Access Memory” $\in I$)

$S = N^S \cup H^H \cup Y^H$ is the set of synonyms, hyponyms and hypernyms of instances of set I .

$N = N^S$ is set of synonyms of instances of set I .

$H = H^H$ is set of hyponyms of instances of set I .

$Y = Y^H$ is set of hypernyms of instances of set I . (e.g., “ADT”, “data structure”, “ADT is a kind of data structure that is defined by programmer” are synonymous, hyponymous and hypernymous of “Abstract data type”)

$B = \{\text{belong_to}(i, c) \mid i \in I, c \in C\}$ is set of semantic relationships between concepts of set C and instances of set I and are denoted by $\{\text{belong_to}(i, c) \mid i \in I, c \in C\}$ mean that i belongs to category c . (e.g., belong_to (“robotic”, “Artificial Intelligent”))

$R = \{\text{rel}(s, i) \mid s \in S, i \in I\}$ is the set of relationships between terms of set S and instances of set I and are denoted by hierarchy and are denoted by $\{\text{rel}(s, i) \mid s \in S, i \in I\}$ mean that s is relationship with i . The relationships can be synonymous, hyponymous or hypernymous. (e.g., synonym (“ADT”, “Abstract data type”), hyponym (“data structure”, “Abstract data type”), hypernym (“ADT is a kind of

data structure that is defined by programmer”, “Abstract data type”). According to Fig. 1, the following sets can be identified as below:

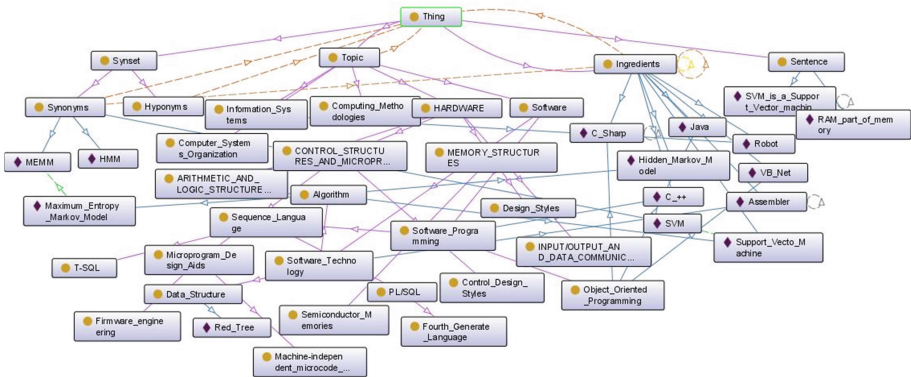


Fig. 1. CDO hierarchy is presented by Protégé

- C = {Software, Software programming, Software technology, Object oriented programming, data structure, Sequence language}
- I = {Abstract data type, Random access memory, Read only memory}
- N = {ADT, RAM, ROM}
- H = {Data structure, database, memory}
- Y = {EPROM, EEPROM, DDRAM, DDRAM2, DDRAM3}

In addition, all concepts and instances of this ontology focus on computing domain; therefore this ontology is known as Computing Domain Ontology (CDO). We separate CDO into four layers:

The first layer is known as the topic layer. In order to build it, we extract terms from ACM Categories [11]. We obtain over 170 different categories from this site and rearrange them in this layer.

Next layer is known as the ingredient layer. In this layer, there are many different instances, which are defined as nouns or compound nouns from vocabulary about Computing domain, e.g., “robot”, “Super vector machine”, “Local Area network”, “wireless”, “UML”, etc. In order to setup this layer, we use Wikipedia to focus on English language and computing domain.

The third layer of CDO is known as the Synset layer. To set up this layer, we use the WordNet ontology. Similar to Wikipedia, we only focus on computing domain. This layer encloses a set of synset. A synset includes synonyms, hyponyms, and hypernyms of instances of the ingredient layer.

The last layer of CDO is known as the sentence layer. Instances of this layer are sentences that represent syntactic relations extracted from preprocessing stage. Hence, these sentences are linked to one or many terms of the ingredient layer. This layer also includes sentences that represent semantic relations between terms of

ingredient layer, such as, IS-A, PART-OF, MADE-OF, RESULT-OF, etc. The overall hierarchy of CDO is shown in Fig. 1.

3.2 The CDO Evaluation

Many techniques are being used for ontology evaluation in the life sciences and more generally. In this section, we look at a number of the techniques: evaluation with respect to the use of this ontology in an information extraction system, with respect to data-driven and the use of data constraints.

Evaluating the Lexicon/Vocabulary and Consistency based on the Data-Driven

Definition 1. *Axioms are the smallest unit of knowledge within an ontology. It can be either TERMINOLOGICAL AXIOM, a FACT or an ANNOTATION. Terminological axioms are either CLASS AXIOMS or PROPERTIES AXIOMS. An Axiom defines the formal relation between ontology entities and their name.*

Definition 2. *Literals are the names that are mapped to concrete data values, i.e. instead of using URI to identify an external entity, literals can be directly interpreted. All of axioms, entities in the Computing Domain Ontology are literals.*

To evaluate the lexicon/vocabulary or axioms, in the first method, we use three measures: Precision (P), Recall (R) and F-measure (F). They are calculated as follows

$$P(C_i) = \frac{Correct(C_i)}{Correct(C_i) + Wrong(C_i)} \quad (1)$$

$$R(C_i) = \frac{Correct(C_i)}{Correct(C_i) + Missing(C_i)} \quad (2)$$

$$F - \text{measure}(C_i) = 2 \frac{Precision(C_i) * Recall(C_i)}{Precision(C_i) + Recall(C_i)} \quad (3)$$

Where C_i represents a category in CDO and correct, wrong, missing represent the number of terms, which are correct, wrong, missing, respectively.

The evaluation of a number of terms, which are wrong or missing in CDO, can only be carried out by validation with respect to other publicly available computing sources. We therefore manually verify the knowledge base with respect to benchmark information provided by the three computing dictionaries as follows:

- Networking dictionary [12] for evaluating the categories related to the network.
- Dictionary of IBM and Computing Terminology [13] for evaluating the categories related to hardware and devices
- Microsoft Computer Dictionary (Microsoft corporation [14]) for evaluating the categories related to software, programming language, etc.

Definition 3. Given an ontology T and a dictionary D , a term I belongs to category C with $C \subset T$, I is called “Wrong” iff there exists a term I' belongs to category $C' \subset D$ such as $I' \equiv I$ with $C' \neq C$.

Definition 4. Given an ontology T and a dictionary D , a term I belonging to category C' with $C' \subset D$, I is called “Missing” iff there exists no term I belonging to T .

We propose the evaluated algorithm for calculating three measures: Precision, Recall, and F-measure as follows

Algorithm 3.1. Calculating three measures: Precision, Recall and F-measure

```

Input: CDO, Networking dictionary, IBM dictionary, Microsoft
dictionary
Output: Precision (Ci), Recall(Ci) and F-measure (Ci)
Precision P ← 0, Recall R ← 0, F-measure F ← 0
Correct ← 0, Wrong ← 0, Missing ← 0
For each category Ci in CDO
  While (instance belong to Ci is not null)
    Case:
      Case 1: Ci belongs to Network category
        Matching instance to network dictionary
      Case 2: Ci belongs to Hardware or Devices
        Matching instance to IBM dictionary
      Case 3: Ci belongs to other categories
        Matching instance to Microsoft dictionary
    End Case
    If (instance is found in the dictionaries)
      Correct = Correct + 1
    Else
      If (instance is found other categories)
        Wrong = Wrong + 1
      Else
        Missing = Missing + 1
      End if
    End if
  End Case
End While
P(Ci) = Correct (Ci)/ Correct(Ci) + Wrong (Ci)
R(Ci) = Correct (Ci)/ Correct(Ci) + Missing (Ci)
F(Ci) = 2 * P(Ci) * R(Ci)/P(Ci) + R(Ci)
P ← 0, R ← 0, F ← 0
End For
Return P, R, F

```

Evaluating the Lexicon/Vocabulary and Consistency based on the Application.

Once again, consistency and vocabulary of CDO are evaluated based on the application. Application-based evaluation offers a useful framework for measuring practical aspects of ontology deployment. The accuracy of responses provided by the system will show the accuracy of the ontology. In case, the application is an information extraction system, which is built based on CDO. The model of this system is shown as Fig. 2.

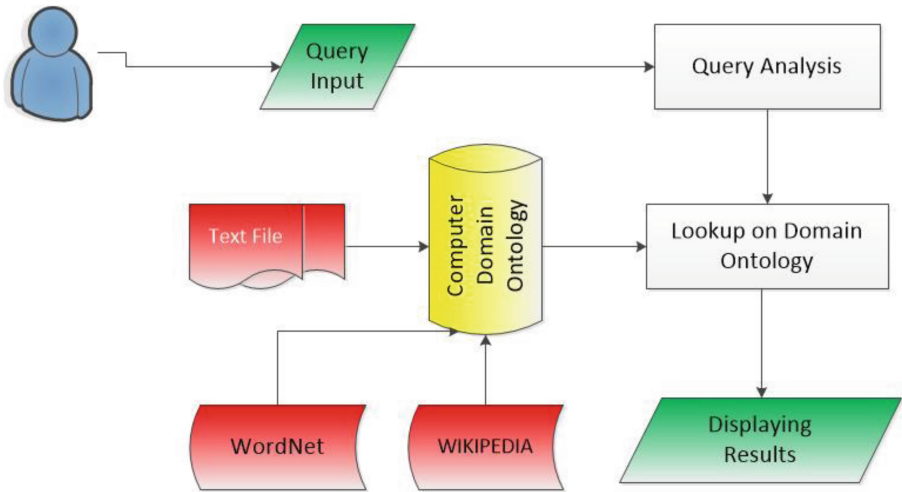


Fig. 2. Model of the information extraction system

According to Fig. 2, the results, which reply to user's queries are filtered and extracted from the different layers of CDO. The accuracy of the results will reflect the accuracy of the terms of CDO and the results will be shown in the next section.

Evaluation of the CDO's Structure and the Relations of Terms. This is primarily of interest in manually constructed ontologies. The structural ontology concerns involve the organization of the ontology and its suitability for application development. There are some of the approaches for evaluating the structure of ontologies. They are:

- Using anti-pattern and heuristic to discover structure errors in ontologies [15]. According to Lam [15], the structure of ontologies is an error because of these anti-patterns.
- Constraint validations in ontologies [4]

Definition 5. *Data constraint is a limitation that was placed on data when it is inserted into ontology.*

In this case, we define some of the data constrains in CDO as follows

- **Instance constraint.** Given a term I belonging to Ingredient or Synset layers, ontology T , set of category C , $I \in T \Rightarrow \exists C_i \in C \mid I \in C_i$
- **Transitive constraint.** Given a term I belonging to Ingredient and Synset layers, ontology T , set of category C with $C_1 \subset C$, $C_2 \subset C_1$, $C_3 \subset C_2$, $I \in C_3 \Rightarrow I \in C_2$, $I \in C_1$, $I \in C$
- **Relational constraint.** Given a relationship $R(I_1, I_2, \dots, I_n)$, set of category C , $C_i \subset C$, $\exists I_i \in R \mid I_i \in C_i$ with $i = 1 \dots n$

The data constraint validations are implemented by Structured Query Language (SQL) since we use Relational Database System (RDBS) for ontological representation. The experiment results will be shown in the next section.

4 Experiment

4.1 Evaluating the Lexicon/Vocabulary and Consistency of CDO Based on Data-Driven

Figure 3 respectively shows the results through three above measures when applying Algorithm 3.1. We choose five categories, which are Hardware, Computer communication network, Network architecture and design, Software engineering, and Programming language for illustration.

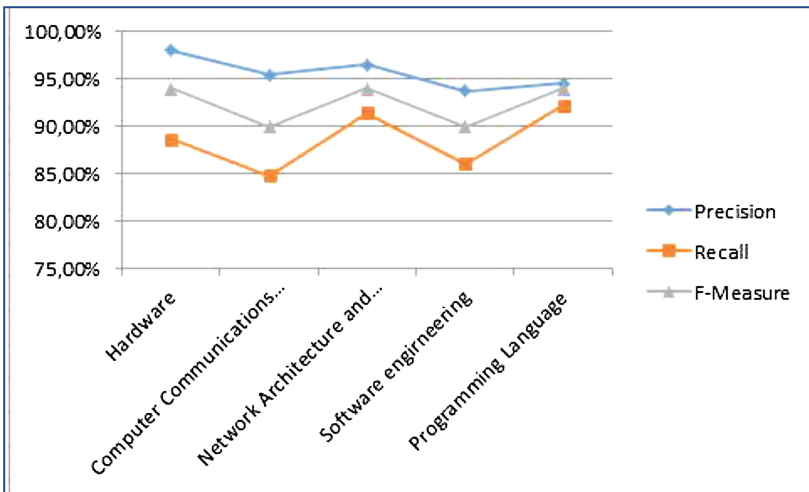


Fig. 3. Evaluation on the lexicon/vocabulary and consistency of CDO based on data-driven

The scores reported in Fig. 3 reveal that the ontological evaluation based on data-driven yields a performance respectably. The Precision measure has a high value. It means that the lexicon/vocabulary of CDO reflects substantially more relevant instances than irrelevant while high recall means that the lexicon/vocabulary of CDO reflects most of the relevant instances.

4.2 Evaluating the Lexicon/Vocabulary and Consistency of CDO Based on Application

As mentioned above, the user's queries, which are inputted directly into the application, are used for application-based evaluation. We also pick the same five categories as first method of illustration. Furthermore, the queries consist of four types of sentences, as follows.

- 80 queries are only noun phrases, e.g., “Java language”, “CPU Pentium”, “Open system internetworking”, etc.
- 80 queries are simple sentences that consist of simple subjects and simple predicates [16]. A simple subject is a noun or noun phrase and the simple predicate is always a verb, verb string or compound verb, e.g., “Java language does”, “CPU Pentium makes”, “Transmission control protocol does”, etc.
- 80 queries are simple sentences that consist of subject and complex predicate [16]. The complex predicate consists of the verb and all accompanying modifiers and other words that receive the action of a transitive verb or complete its meaning, e.g., “Java is programming language”, “What is transmission control protocol”, “Transport layer provides services to the Network layer”, etc.
- 80 queries consist of complex sentences, wrong grammar sentences, and unfinished sentences, i.e. they do not contain a complete idea, e.g., “control transmission protocol”, “table routing”, “Mac Address is a unit address for a computer belongs to Data Link layer”, etc.

Figure 4 respectively shows the results.

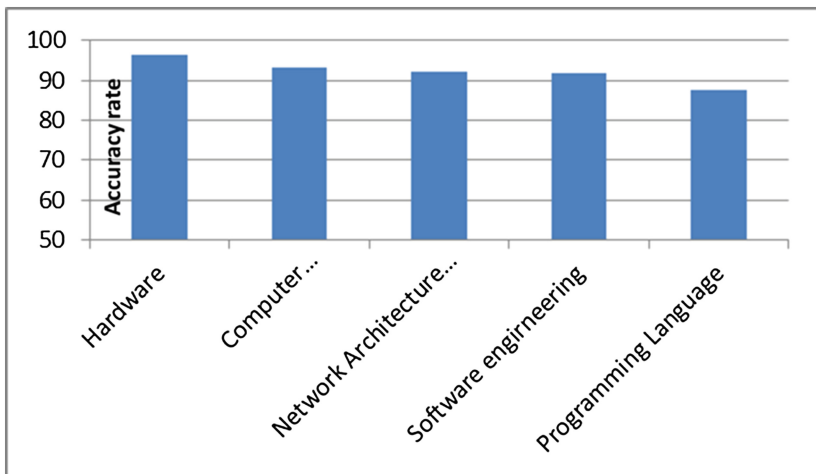


Fig. 4. Evaluation on the lexicon/vocabulary and consistency based on the application

The scores reported in Fig. 4 reveal that the accuracy rates of the results are returned from the information extraction system when the system extracts information

related to the different queries (96 % for hardware category and 87 % for programming category).

4.3 Evaluating the CDO's Structure and the Relations of Terms

As mentioned above, we use SQL for data constraint validations. Some of the SQL scripts are written for validations. We also pick five categories as previous sections for illustration. Figure 5 respectively shows the results.

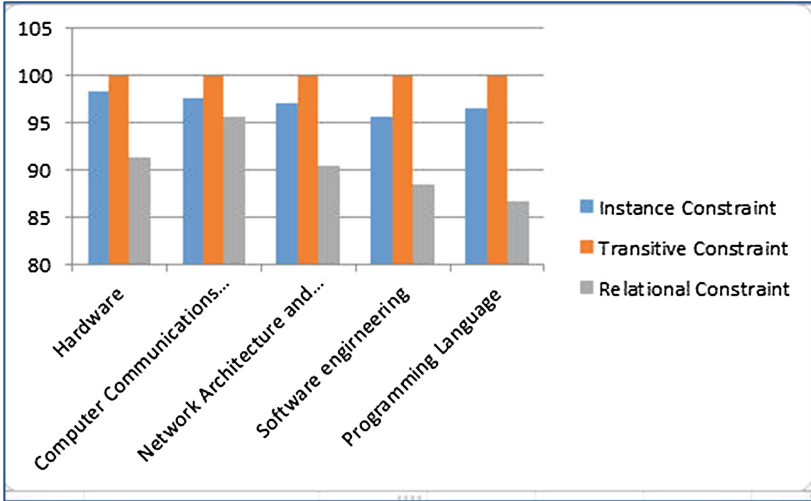


Fig. 5. Evaluation on the ontological structure and the relations of terms

As the above-mentioned definitions, the scores reported in Fig. 5 reveal that the minimum of the instance constraint reaches about 91 %; it means that all instances of the synset layer have a relationship with at least an instance of ingredient layer. Moreover, the transitive constraint reaches 100 %; it means that the structure of CDO is reasonable while the minimum of relational constraint reaches about 82 %.

5 Conclusions

In this paper, we dealt with the problem of the ontological evaluation. This ontology focuses only on computing domain and it has the complex structure. In order to evaluate the lexicon/vocabulary or axioms and consistency of the ontology, we proposed two methods; (i) based on data-driven; (ii) based on information extraction system. We also used the data constrains for validation of the ontological structure and the relations of terms. Results generated by such experiments show that the terms and axioms belonging to different layers of the ontology have a high accuracy rate and the ontology can be used for many different applications, such as, Information Retrieval applications, Information Extraction applications. Comparing to other frameworks for

evaluating the ontologies, such as, OntoClean, OntoManager, OntoMetric, our proposed approach evaluates the lexicon/vocabulary of CDO based on not only the data-driven, but also the application and data constraint in order to check the accuracy of instances of CDO.

In the future work, we will focus particularly on automatically ontological enriching, but the accuracy rate in terms of the ontology is still high. Besides, the data constraint validations are also satisfied.

References

1. Pérez, A.G.: Some ideas and examples to evaluate ontologies. In: Proceedings of the 11th Conference on Artificial Intelligence for Applications, pp. 299–305, Los Angeles, CA (1995)
2. Fahad, M., et al.: A framework for ontology evaluation (2008). <http://ceur-ws.org/Vol-354/p59r.pdf>
3. Velardi, P.: Evaluation of OntoLearn, a methodology for automatic learning of domain ontologies, The Pennsylvania State University. <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.97.333&rep=rep1&type=pdf>. Accessed 9 September 2015
4. Guarino, N., et al.: An overview of OntoClean. https://noppa.aalto.fi/noppa/kurssi/as-75.4700/materiali/AS-75_4700_overview_of_ontoclean.pdf. Accessed 9 September 2015
5. Stojanovic, N., et al.: The OntoManager – a system for the usage-based ontology management. <http://www.kde.cs.uni-kassel.de/ws/LLWA03/fgml/final/Stojanovic.pdf>. Accessed 9 September 2015
6. Tello, A.L., et al.: ONTOMETRIC: a method to choose the appropriate ontology, archivo digital UPM. http://oa.upm.es/6467/1/ONTOMETRIC_A_Method.pdf. Accessed 9 September 2015
7. Netzer, Y., Gabay, D., Adler, M., Goldberg, Y., Elhadad, M.: Ontology evaluation through text classification. In: Chen, L., Liu, C., Zhang, X., Wang, S., Strasunskas, D., Tomassen, S. L., Rao, J., Li, W.-S., Candan, K.S., Chiu, D.K.W., Zhuang, Y., Ellis, C.A., Kim, K.-H. (eds.) WCMT 2009. LNCS, vol. 5731, pp. 210–221. Springer, Heidelberg (2009)
8. Soysal, E., Cicekli, I., Baykal, N.: Design and evaluation of an ontology based information extraction system for radiological reports. *Comput. Biol. Med.* **40**(11–12), 900–911 (2010)
9. Obrst, L., et al.: The evaluation of ontologies. In: Baker, C.J.O., Cheung, K.H. (eds.) *Semantic Web: Revolutionizing Knowledge Discovery in the Life Sciences*, (chap. 7), pp. 139–158. Springer, New York (2007)
10. Zhang, L.: Ontology based partial building information model extraction. *J. Comput. Civ. Eng.* **27**, 1–44 (2012)
11. Association for Computing Machinery. <http://www.acm.org/about/class/ccs98-html>. Accessed 9 September 2015
12. Sybex corporation (2000). www.sybex.com
13. IBM. <http://www-03.ibm.com/ibm/history/documents/pdf/glossary.pdf>. Accessed 9 September 2015
14. Microsoft corporation. <https://robot.bolink.org/ebooks/Microsoft%20Computer%20Dictionary%205e.pdf>. Accessed 9 September 2015
15. Lam, J.: Methods for resolving inconsistencies in ontologies. Ph.D. thesis, University of Aberdeen, Aberdeen, Scotland (2007)
16. Capital Community College Foundation. <http://grammar.ccccommnet.edu/grammar/objects.htm>. Accessed 9 September 2015

Data Models and Advances in Query Processing

Comics Instance Search with Bag of Visual Words

Duc-Hoang Nguyen^{1,2(✉)}, Minh-Triet Tran¹, and Vinh-Tiep Nguyen¹

¹ Faculty of Information Technology, University of Science,
VNU-HCM, Ho Chi Minh City, Vietnam
hoangnguyen@squarebitinc.com,
{tmtriet, nvtiep}@fit.hcmus.edu.vn

² Squarebit Inc., Ho Chi Minh City, Vietnam

Abstract. Comics is rapidly developing and attracting a lot of people around the world. The problem is how a reader can find a translated version of a comics in his or her favorite language when he or she sees a certain comics page in another language. Therefore, in this paper, we propose a comics instance search based on Bag of Visual Words so that readers can find in a collection of translated versions of various comics with a single instance as a comics page in an arbitrary language. Our method is based on visual information and does not rely on textual information of comics. Our proposed system uses Apache Lucene to handle inverted index process to find comics pages with visual words and spatial verification using RANSAC to eliminate bad results. Experimental results on our dataset with 20 comics containing more than 270,000 images achieve the accuracy up to 77.5 %. This system can be improved for building a commercial system that allows a reader easily search a multi-language collection of comics with a comics page as an input query.

Keywords: Visual instance search · Comics · Bag of visual words · Lucene

1 Introduction

Nowadays, comics is developing rapidly and strongly all around the world. It is attracting the attention of a lot of people at many places and in various ages. This is proven by the growth of the comics book industry. In some countries, such as Japan or US, comics has become a part associated with its culture and social. In these countries, comics is separated to special types with the special name indicated for the comics at there. In particular, Manga is used for comics of Japanese or American Comics is used for comics of US. DC and Marvel are the biggest comics publishers that many film is adapted from their products. In the end of 2014, One Piece, a famous Manga, has 320,866,000 printed copies worldwide [1].

With the rapid developing of comics worldwide, a comics can be translated to different languages. For example, One Piece has already officially published in over 30 countries [1]. When someone sees a comics page on a website or on a shared post from social network, she might want to know that where she can read it or if it has a translated version in her favorite language. In this case, if using textual information, she

will easily find out the name of the comics. However, it is really difficult to find where a specified page exactly come from in that comics.

Furthermore, it also not easy to find a translated version of a given page in an arbitrary language even if we can know the original name of the chapter containing the considered page. Actually, the translated chapter name is usually different due to given by idea only. For example, a name of One Piece chapter in English is “Romance Dawn” but in Vietnamese is “Bình minh của cuộc phiêu lưu”, means dawn of adventure. Besides, some translated comics come with different number of chapters in a volume, even number of pages in a chapter. Some electronic versions of a specified comics can contain different structure from the corresponding printed versions. So, some versions have no correspondence to the original one. Besides, the text in comics do not always appear separately from the figures, the text is usually mixed into figures and can be in different fonts and sizes, even appears with decoration in a figure. So, understanding the content of a comics from text only is not an easy task. Therefore, it motivates us to use the visual information of image to solve this problem.

Using the visual information also encounters some problems. Each comics is translated into different languages with different versions and different variants. A specified language can produce many translated versions of a comics (Fig. 1a) because a comics when is translated into a given language can be made by many groups of translators: it can be translated by some groups of volunteers or by some official publishers. Therefore, this makes variety of translated versions. Besides, variants of a comics are also created because of some reasons. First, it is depending on the reading style. Particularly, Japanese reading style is back to front and right to left from a book and verbose for reading style of Vietnamese and others, it makes a variant type that flipping images vertically (Fig. 1b). Next type of variant is some part of the comics might be modified to adapt with the culture, law, or social view point. (Fig. 1c). Another variant of comics can be produced after translation when texts appear as decoration of page and provide signification differences in visual perception. They can see easily by human eyes but in computer eyes, they are completely different features (Fig. 1d). With all reasons mentioned above, they make some difficulties to find some other versions of the specified image in a large dataset of pages in different translated versions and variants of them.

In this paper, we propose a system to search for translated versions and variants of a specified comics page based on a visual instance of that comics page. The input data of this system is a comics page in an arbitrary language, the system finds the corresponding pages in different translated versions and variants of the input query. By this way, readers can easily find a translated version corresponding to a given page or a chapter that she wants to read in her mother language. Our proposed solution is based on visual instance search framework with Bag of Visual Words [10].

Main Contribution First, we conduct an experiment to find the type of features suitable to represent the characteristic of comics images, and we find that ORB [5] is a solution that has better performance than SIFT [3] and SURF [4] and achieves competitive accuracy against SIFT and SURF. Second, we propose and implement an instance search system for comics based on Lucene, a strong search engine used widely in software industry. We also apply RANSAC as the post processing step for spatial

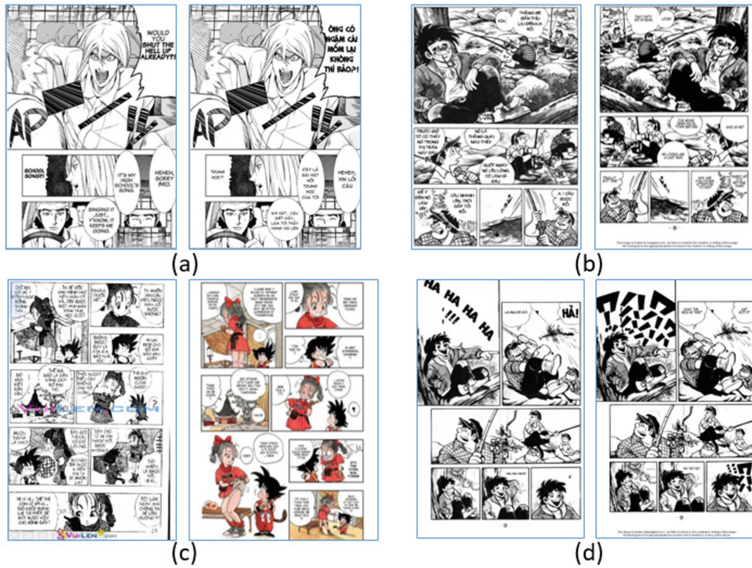


Fig. 1. Some significant problems of finding corresponding images using visual information: (a) original page and its different language version; (b) original page and its modified version with flipping vertically; (c) original page and its modified version with omitted or modified parts; (d) original page and its modified version with different decorative texts.

verification in order to push the list of results and eliminate bad matching candidates. Figure 2 demonstrates a query and produced results of system. Last, we also conduct an experiment for evaluating the number of visual words that is the best for using in this system. The system is evaluated with more than 270,000 pages of 20 different comics in different languages and the result of 750 test queries achieves the accuracy up to 77.5 %.



Fig. 2. A sample of query input and search results

The content of this paper is as follows. In Sect. 2, we briefly review different methods for visual instance search. Our proposed method for comics instance search is presented in Sect. 3. Section 4 is for experiments and evaluation. The conclusion and discussion for future work are in Sect. 5.

2 Background

The original Bag of visual words (BOW) model, introduced by Sivic for video retrieval [10], is a foundation for most of the state-of-the-art image retrieval systems. This model bases on the sharing significant number of local patches, called the key assumption, of two similar images. Sparse feature detectors, such as DoG [11], Hessian-Affine [12] or MSER [13], is very efficient to detect interesting regions of rich-textured objects such as buildings, paintings, advertising posters.

Many techniques, such as RootSIFT feature [14], large vocabulary [15], soft assignment [16], multiple detectors and features combination at late fusion [17] or query-adaptive asymmetrical dissimilarities [18], have been proposed to improve the performance of retrieval systems. Among these methods, spatial verification is one of the most effective approaches, and is also used as a preprocessing step for other advanced techniques. Spatial verification can be classified into two classes: spatial reranking and spatial ranking.

Spatial reranking checks the geometric consistency of visual words on a short list of about 200 to 1000 results given by the BOW model. An effective solution for this problem is using RANSAC and exploiting the local shape of the affine covariant region for rigid affine consistency checking, which was first applied by Philbin et al. [15]. Hough Pyramid Matching approach for spatial reranking uses a hierarchical structure to group matches, thus resulting in an algorithm which is only linear in the number of putative correspondences [19]. Also, an elastic spatial checking technique was proposed to emphasize the topology layouts of the matching points [20].

Spatial ranking incorporates the spatial information at the original ranking stage to improve the efficacy of the search system. Jegou et al. [21] introduced a method using a Houghlike voting scheme in the space of similarity transformation between the query and training images, but this is just a weak geometric consistency checking (WGC). Cao et al. proposed to use spatial-bag-of-features which capture the spatial ordering of visual words under various linear and circular projections [22]. Shen et al. proposed to transform the query ROI by the predefined scales and rotations [23]. However, this method is much more expensive in cost of computation than other approaches such as BOW or WGC.

3 Proposed Comics Instance Search System

Our proposed system includes two stages: training and search. The training stage is responsible for providing the data indices of training images used to search similar images from a query on search stage. Illustrated by Fig. 3, the training stage of this system comes with four main steps:

- **Step 1:** Comics images in database are extracted features with ORB algorithm [5]. This step produces rBRIEF descriptors and key points.
- **Step 2:** We will pick randomly n descriptors for training the k -means [2] model. As a result of this step, we have a collection of k -means vectors that each of them is the center point of k clusters.

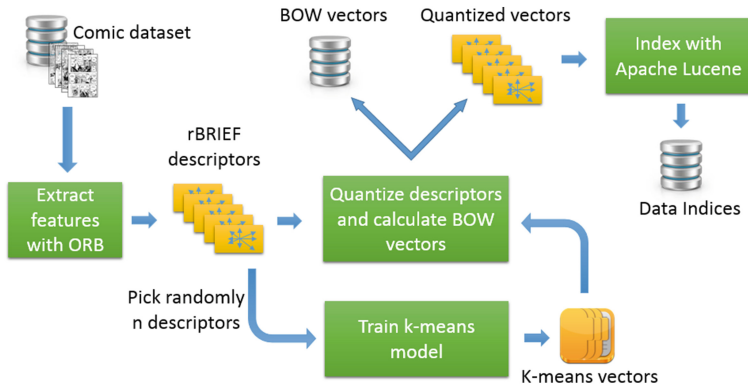


Fig. 3. Proposed training system

- **Step 3:** All rBRIEF descriptors are quantized using k-means vectors. This step produces quantized vectors and BOW vectors.
- **Step 4:** The quantized vectors will be indexed using Apache Lucene search engine library.

The final results of this training stage are k-means vectors, BOW vectors and data indices used for search stage.

As the name of the stage, search stage is a procedure for searching a query image. The result of this stage should be a ranked list of similar trained images. Figure 4 demonstrates the five main steps of system for searching a query image:

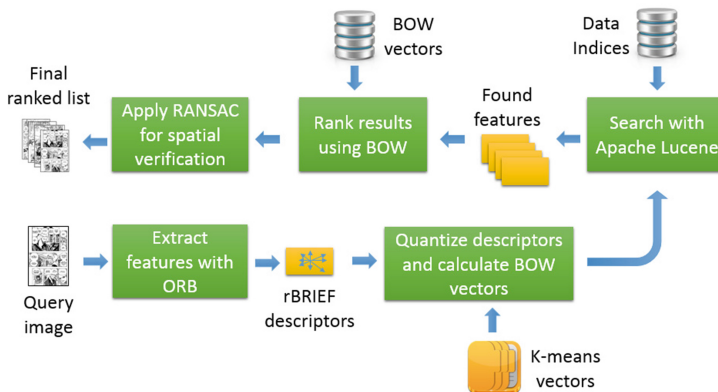


Fig. 4. Proposed search system

- **Step 1:** The query image also will be extracted features with ORB algorithm.
- **Step 2:** The rBRIEF descriptors of this image is used to calculate labels string with k-means vector obtained from training stage.

- **Step 3:** This labels string is used as a query input for Apache Lucene search engine. As a result of this step, we obtain a list of found items with their features.
- **Step 4:** We rank the results by comparing the distance from the BOW vector of query image to BOW vectors of found items. Finally, the ranked list of results is obtained.
- **Step 5:** As the post processing step, we apply the spatial verification using RANSAC. This step produces a high quality results after eliminating bad results.

In this system, consider to persistent information storage, we need store 3 addition type of data beside comics dataset. They are k-means vectors, BOW vectors, search engine indices. Following parts will describe modules of system in detail.

3.1 Image Feature Extraction with ORB

Feature extraction used in both training and search stage of this system is a important module in this system. Because the output data of this module are used in all others modules and they are foundation of performance, efficiency and accuracy of the entire system. In this system, we use ORB (Oriented FAST and Rotated BRIEF) algorithm.

ORB is a robust feature extracting algorithm introduced by Ethan Rublee, et al. at ICCV 2011 [5]. As the title of proposed paper, this algorithm is considered as a good alternative to SIFT (Scale-Invariant Feature Transform) [3] and SURF (Speeded-Up Robust Features) [4] in computation cost and matching performance. Besides, it is a child of “OpenCV Lab” so it is strongly and fully supported from OpenCV library. ORB is a combination of FAST (Features from Accelerated Segment Test) key point detector [6] and BRIEF (Binary Robust Independent Elementary Features) descriptor [8] with many modifications to enhance the performance.

In detail of ORB, first it find keypoints using FAST algorithm, then find top result using Harris corner measure and apply pyramid to produce multiscale features. However FAST doesn’t compute the key point orientation, so it considers a patch of pixels with key point as the center and computes the intensity weighted centroid of the patch. The direction of the vector from the center point to intensity centroid provides the orientation of the key point. Second, ORB uses BRIEF for feature descriptors. But BRIEF is not good for describing rotation, so ORB performs some modifications to solve this problem. The result is called Rotated BRIEF or rBRIEF. For matching rBRIEF descriptors, ORB uses multi-probe LSH, an improvement of the traditional LSH. In conclusion, the paper says ORB is faster than SURF and SIFT on extracting and ORB is better than SURF on matching.

We choose ORB because of four reasons: comics image characteristic, storage size efficiency, processing performance and matching accuracy. About characteristic of image, comics images contain a lot of strokes, this makes SIFT and SURF really slow because they must process too many key points (Fig. 5 gives a sample for this). Instead, in our practice, ORB handles well this problem with high performance and acceptable accuracy. About storage size efficiency, rBRIEF uses only 32 bytes for a descriptor while SIFT uses 512 bytes for a 128-dim vector descriptor and SURF uses 256 bytes for a 64-dim vector descriptor. In our experiment, ORB is 24.65 times faster than SIFT

and 4.987 times faster than SURF in the average processing time of feature extracting. As a result, ORB completely wins SIFT and SURF on processing performance. Also in our experiment, the accuracy on matching of ORB competes to SHIFT and SURF with passing more than 90 % of tests. Put all reasons in place, ORB is a reasonable choice for feature extraction.

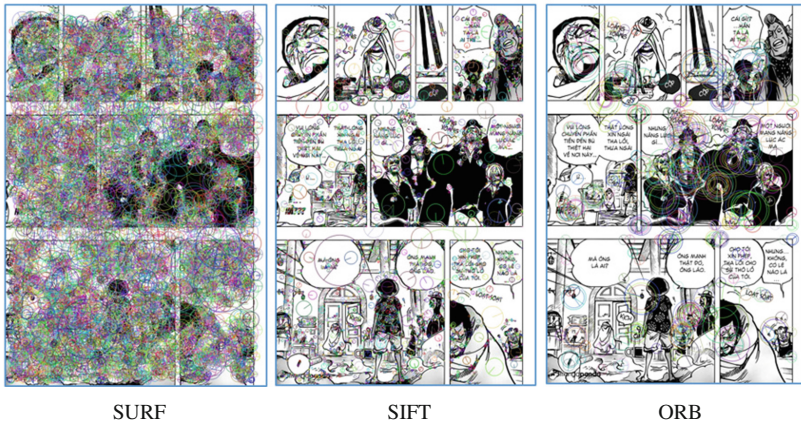


Fig. 5. Feature extraction in a comics page using SURF, SIFT, ORB features

3.2 Train K-means Model by RBRIEF Descriptor Vectors

K-means [2] is one of the simplest algorithms of unsupervised machine learning for solving the clustering problem. This algorithm is an iterative process to partition n vectors into k clusters.

In our system, k-means is used to cluster the rBRIEF descriptor vectors obtained from ORB feature extraction. This clustering results, a very important data, will be used for indexing image features and searching similar images. Therefore, they decide the accuracy of system. In this step, we pick randomly a collection of 1 million vectors in all descriptors extracted from comics image dataset and train k-means model to obtain k cluster center vectors. In our experiment, we find out $k = 30,000$ gives the best result for searching step.

3.3 Descriptor Vector Quantization

This module is responsible for applying quantization to feature descriptors of an image and producing a quantized vector and BOW vector of the specified image. A quantized vector is a vector that each element is the index of cluster that a corresponding descriptor belong to. A BOW vector is a histogram of the cluster frequency of descriptor quantization. This histogram has k bins where k is the number of k-means clusters. This histogram can be built in hard assignment way that means for each visual word we just vote only one for the best cluster. However, in order to improve the

accuracy of our system, we apply the soft assignment. Each feature can vote to different relevant clusters according the distance from that feature to the centroid of clusters. In our experiment, we choose the number of relevant clusters used for soft assignment is three.

With each of feature descriptor in rBRIEF descriptors set of image, we find the cluster of k-means clusters to which this descriptor belongs by calculating the nearest Euclidean distance of considering descriptor vector and k-means center vectors. Actually for soft assignment, we will find 3 nearest clusters for a description. Index of the best cluster will be assign to corresponding element of quantized vector. Then, we vote 0.6 for bin of best cluster in BOW vector and 0.2 for bins of remaining clusters. This process is called as vector quantization. The results of this step are quantized vector and BOW vector of the specified image.

3.4 Pattern Indexing and Searching with Apache Lucene

The objective of this step is finding a set of images in dataset which related to the query image. An image is considered to be related if it has at least one the visual keyword contained in the query image. A common solution is using the invert index tree that the key of map is a visual word and the value of a key is a linked list of image containing the corresponding visual word as mapping key. Instead of building this invert index tree manually, we use a professional tool, Apache Lucene, to handle this.

Because Lucene is a text search engine, we must convert the quantized vector to a string of base 16 words converted from each element of quantized vector. This label string will be used for indexing and searching images using a text search engine that each token for indexing corresponds to a label of each feature descriptor.

Apache Lucene is a very strong library for text search that widely used in many large systems. As an example, Apple, Disney Twitter websites are using Lucene for their real time search [25], and Elasticsearch server is based on Lucene. [24].

Apache Lucene is a free open source text search engine library [25] written first in Java by Doug Cutting. It is a technology suitable for application that requires full text indexing and searching capability with high-performance and full-featured. Doug Cutting originally wrote it in 1999. Then, it joined to Apache Software Foundation's Jakarta family of open source Java products in 2001 and became top-level Apache project in February 2005. The latest version of Lucene is 5.2.0 which released on June 7, 2015. It also has been ported to many programming languages as Delphi, Perl, C#, C++, Python, Ruby, and PHP [24].

Apache Lucene includes five main modules as Document, Analyzing, Storing, Indexing and Searching. First, at the core of logical architecture is idea of a Document containing Fields, whose values may be strings or instances of content. Second, Analyzing module provides analyzers used for converting raw text to tokens that will be actually indexed. There are some build-in analyzers included in library such as Standard Analyzer for grammar based analysis and Whitespace Analyzer for whitespace based separation. Third, Storing module defines location for storing persistent data that include indices and associate data. Forth, Indexing module is used to create and add documents to indices and to read indices from stored places. For these features,

Index writer and Index reader are two main components in this module. Last, Searching module provides data structures to represent queries and finds out top documents in indexed data. Index Searcher is the major component in this module.

In order to create indices, Documents, which should consist of at least one Field of content, are analyzed and converted to tokens by an Analyzer. After that, the Index Writer coordinates things to get a meaningful index and creates indices. These indices will be stored in a Directory which is a component of storing module. Directory may be a real directory in file system or a virtual place in memory. Put these things in place and we have the Lucene Index procedure.

In order to search an index, the same analyzer used in indexing procedure is reused to analyze the query document to query tokens. Then, a Query is built from these analyzed tokens and an Index Searcher is used to run query on the indices stored in the Directory. The result of this execution is a list of documents found for the searching terms. Each result in list includes the corresponding document identifier and the score that indicates how good matching it is. Put all of them together, we have the Lucene Search procedure.

In case of our system, we must ensure that each clustering label of image descriptor should be indexed by search engine. Therefore, the most suitable analyzer is Whitespace Analyzer. Beyond this note, all of procedures for indexing or searching of our system are the same as Lucene workflow.

3.5 Ranking the Results Using BOW Vectors

The previous step gives us a collection of related image corresponding to the query. As the final step of this system, a ranking process is applied to find the most similar images as the final results.

The objective of this module is ranking the results given by searching the relevant images in dataset. We rank the results according to the similarities of these candidates with the given query. In particular, we calculate the Euclidean distance of the BOW vectors of candidates and BOW vector of the query image and select top N nearest items as the result.

In the future, they can be different formulas and methods in order to rank the results. But currently we just use the Euclidean distance to estimate the difference between the BOW vectors for this ranking task.

3.6 Spatial Verification Using RANSAC

This step is used for eliminating bad results. This module is an acceptant test for selecting eligible results. A verifying image should be accept only if it matches the query image. This step will improve the quality of results.

The key idea of matching decision is that a pair of images will be matched only if there are at least T matched keypoint pairs which pass an homography test. In other words, if we can find a perspective transformation matrix that has ability to transform T points from the query image to the verifying image so that each transformed point

approximately matches the corresponding point of verifying image in the considering keypoint pair, we can decide that two images are matched.

For calculating the homography matrix, we use the RANSAC (Random Sample Consensus) algorithm [9]. It provides an estimation for a mathematical model with input as a set of outliers containing data. The objective of this algorithm is producing a reasonable result only with a certain probability that more iterations should increase this probability.

By this way, we will eliminate candidates unsatisfying the spatial verification step. This processing pushes the accuracy of final results. As the characteristic of comics images, they only can be changed by 2D transformations but not 3D projections, so applying spatial verification using RANSAC is a logical approach.

Going into detail, in this step, we have a list of image results from search step. Now, each result image given will be applied these steps for verify if it should be accept:

- **Step 1:** Find matched keypoint pairs between verifying image and query image using Brute-Force matching.
- **Step 2:** Pick top N best score pairs and apply RANSAC to calculate perspective transformation matrix for these pairs. In experiment, we choose $N = 100$ and call the transformation matrix as M .
- **Step 3:** Transform one by one matched points from query image to verifying image using M perspective matrix and verify the transformed point if its distance to corresponding matched point of verifying image is less than a threshold D pixels. In experiment, we choose $D = 10$.
- **Step 4:** Count passing results of previous step. If number of them is greater than a threshold T , the system decide that two images match together. Otherwise, they are unmatched. In practice, we find out $T = 30$ gives the best decisions.

In practice, our experiment applies this method to features extracted from test comics images by SIFT, SURF and ORB. The results of all algorithms are good with more than 90 % tests passing. In conclusion, this method is suitable for verify if a pair of comics images are matched.

4 Experiments

We conduct two experiments. First, we do an experiment to verify performance and accuracy of the chosen feature extraction algorithm, ORB, compared to SIFT and SURF algorithms in both two phases of features extracting and matching. Second, an experiment evaluates the best value of k used in k -means clustering step. This experiment is executed on full-feature of proposed system using vary values of k and measures the accuracy of search stage to evaluate the best k . The first experiment will proof the reasonability of chosen algorithm for features extraction, and the second one will provide the best parameters for proposed system as well as the accuracy of entire system.

4.1 Ability of ORB Compared to SIFT and SURF

This experiment is used to test the performance and accuracy of three feature detector algorithm: SIFT, SURF and ORB in the dataset of comics images that each image contains almost strokes.

Test Environment and Dataset For the test environment, we execute this experiment on a system with hardware configuration: Intel Core i3 2.53 GHz CPU and 4 GB of RAM.

The objective of this experiment is measuring to find out an algorithm with fast speed and high accuracy. So we build and ground truth a test dataset to benchmark the performance and accuracy of test algorithms. Particularly, the dataset of this experiment includes 2000 pairs of comics images and it grounded truth. In detail, the dataset has 750 matched pairs and 1250 mismatched pairs. In these mismatched pairs, there are 50 % pairs of the same comics that they have the same style and 50 % pairs of different comics that they have completely different styles. Because of our requirement that this system can be make some wrong rejects but should not be make wrong accepts as a requirement, we uses a test dataset that mismatched test cases (62.5 %) are more than matched test cases (37.5 %).

Experiment With each test algorithm, we execute two phases: feature extracting and matching. The matching phase is introduced in part F of Sect. 3.

Before running this experiment, we apply some modify parameters to tested algorithms. First, because of the characteristic of comics images that contains almost stroke, SURF will find and process too many detected keypoints with default settings of OpenCV. It make SURF very slow, even slower than SIFT. For solving this, we use 15000 as the Hessian threshold for SURF. In practice we find that this parameter is good for SURF handling comics images with high performance and accuracy. Second, for ORB, we use 1000 as the threshold of top features. We also find that this parameter is really good for ORB processing comics images.

As a result, the performance comparison is introduced by Table 1. About speed, the average features extraction speed of ORB is 24.65 times faster than SIFT and 4.987 times faster than SURF. Besides, the average matching speed of ORB is also faster than both SIFT and SURF with 25.9 times and 1.516 times. In the comparison of speed, ORB is the best of all.

Table 1. Performance comparison between SIFT, SURF and ORB

	Average		
	SIFT	SURF	ORB
Extracting speed (in seconds)	2.161	0.437	0.088
Matching speed (in seconds)	3.605	0.211	0.139
Storage size (in KB)	3702.14	520.05	31.25

In big dataset, the size of storing data will seriously affect to the system. Table 1 provides the average storage size of three algorithm. In particular, SIFT is 118.5 times

bigger than ORB and SURF is 16.64 times bigger than ORB. So, we can see that ORB is the best choice for storage size efficiency.

Finally, Table 2 describes the accuracy of three algorithms. As a result, all three algorithms are competing on accuracy with over 98 % correct results in 2000 test image pairs. It's a high accuracy. In detail, ORB has 13 failed tests (99.35 %), SURF has 12 failed tests (99.4 %) and SIFT has 23 failed tests (98.85 %).

Table 2. Accuracy comparison between SIFT, SURF and ORB

	Total		
	SIFT	SURF	ORB
Number of unsuccessful cases	23	12	13
Accuracy (%)	98.85	99.40	99.35

Put all in place, with high processing performance, storage size efficiency and competitive accuracy, ORB is a reasonable choice for feature extraction.

4.2 Determine the Number of Visual Words

This experiment is used to estimate the number of clusters or visual words. In other words, the objective of this experiment is evaluating the best value of k for k -means clustering.

Dataset The dataset of this experiment includes 136,506 images of 20 different comics in different languages. Each image in dataset includes original version and its flip version for handling the case of flip pages, so dataset actually has 271,512 images total.

About test queries, we build and ground truth 750 tests. Each test includes a pair of images: query image and expect image. All query images are not in training dataset and corresponding expect image is contained in the dataset. A test will pass if the expect image is in top ten search results of the associated query image. Query images and expect images are in different languages and in different visual perceptions.

Experiment First, we extract features of all images in dataset using ORB. Then we pick randomly 1,000,000 features among them for training k -means clustering model. In the query procedure, we apply the spatial verification using RANSAC as the post processing step to push the quality of results. We execute the experiment with values of k from 100 to 30,000 and run the test queries. The accuracy is evaluated by checking if the expect image of a specified query is in top ten results of search system. Figure 6 provides the charts of the accuracy and the processing time of search process with vary values k from 100 to 30,000. And Table 3 introduces full results of this experiment.

In conclusion based on experiment results, for accuracy of search system, we find out the k value = 30,000 giving the best result for searching step and the accuracy can be increase if we increase value of k . However, increasing the value of k will affect to the performance of system. We can see that the accuracy doesn't increase so much with high values of k but the processing time increases quite a lot. For the balance between

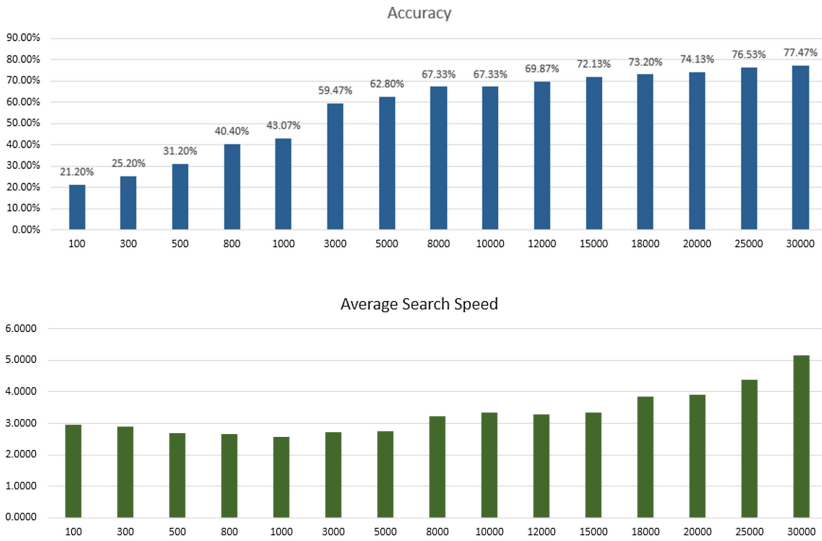


Fig. 6. Accuracy in percent and average processing time in seconds corresponding to numbers of visual words

accuracy and performance, in this system, we choose the 30,000 as value of k with the average search speed about 5 s, an acceptable speed for an image search system.

Table 3. Experiment results by values of k

K	Accuracy in percent	Average Search Speed
100	21.20 %	2.9617
300	25.20 %	2.8953
500	31.20 %	2.6762
800	40.40 %	2.6673
1000	43.07 %	2.5538
3000	59.47 %	2.7145
5000	62.80 %	2.7570
8000	67.33 %	3.2253
10000	67.33 %	3.3503
12000	69.87 %	3.2969
15000	72.13 %	3.3445
18000	73.20 %	3.8375
20000	74.13 %	3.9070
25000	76.53 %	4.3969
30000	77.47 %	5.1703

5 Conclusion

We propose a search system to find comics based on visual information. Using this system, comics readers can easily find a translation of a comics page in their desired language and even they can find different versions of translation in a certain language just based on the visual information. The proposed method can overcome the difficulties to recognize the textual information from the pictures in each page of the comics because of the decoration of the texts and the mixture of pictures and texts. We have already implemented the prototype of system using Lucene to build inverted index of comics pages by visual words, Bag of visual word to rank the results given by Lucene with dataset of over 270,000 images and the spatial verification using RANSAC to eliminate bad results as the post processing step. As the result of experiment, the system takes average about 5 s for a search query and reaches about 75 % of accuracy. This is appropriate for a real application with further optimizations. In addition, we are continuing to try different methods to effectively build and compare histogram vectors.

References

1. One Piece Manga sets Guinness World record (in English). Anime News Network. <http://www.animenewsnetwork.com/news/2015-06-14/one-piece-manga-sets-guinness-world-record-for-copies-printed-for-comic-by-single-author/.89275>. Accessed 15 June 2015
2. MacQueen, J.B.: Some methods for classification and analysis of multivariate observations. In: Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability, vol. 1, pp. 281–297. University of California Press, Berkeley (1967)
3. Lowe, D.G.: Object recognition from local scale-invariant features. Proc. Int. Conf. Comput. Vis. **2**, 1150–1157 (1999)
4. Herbert, B., Andreas, E., Tinne, T., Luc, V.G.: SURF: speeded up robust features. Comput. Vis. Image Underst. (CVIU) **110**(3), 346–359 (2008)
5. Ethan, R., Vincent, R., Kurt, K., Gary R.B.: ORB: an efficient alternative to SIFT or SURF. In: ICCV, pp. 2564–2571 (2011)
6. Edward, R., Tom, D.: Machine learning for high speed corner detection. In: 9th European Conference on Computer Vision, vol. 1, pp. 430–443 (2006)
7. Edward, R., Reid, P., Tom, D.: Faster and better: a machine learning approach to corner detection. IEEE Trans. Pattern Anal. Mach. Intell. **32**, 105–119 (2010)
8. Calonder, M., Lepetit, V., Strecha, C., Fua, P.: BRIEF: binary robust independent elementary features. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part IV. LNCS, vol. 6314, pp. 778–792. Springer, Heidelberg (2010)
9. Martin, A.F., Robert, C.B.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM **24**(6), 381–395 (1981)
10. Sivic, J., Zisserman, A.: Video Google: a text retrieval approach to object matching in videos. Proc. Int. Conf. Comput. Vis. **2**, 1470–1477 (2003)
11. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60**(2), 91–110 (2004)
12. Mikolajczyk, K., Schmid, C.: Scale & affine invariant interest point detectors. Int. J. Comput. Vis. **60**(1), 63–86 (2004)

13. Extremal, M.S., Matas, J., Chum, O., Urban, M., Pajdla, T.: Robust wide baseline stereo from. In: In British Machine Vision Conference, pp. 384–393 (2002)
14. Arandjelovic, R., Zisserman, A.: Three things everyone should know to improve object retrieval. In: IEEE Conference on Computer Vision and Pattern Recognition (2012)
15. Philbin, J., Chum, O., Isard, M., Sivic, J., Zisserman, A.: Object retrieval with large vocabularies and fast spatial matching. In: IEEE Conference on Computer Vision and Pattern Recognition (2007)
16. Philbin, J., Isard, M., Sivic, J., Zisserman, A.: Lost in quantization: improving particular object retrieval in large scale image databases. In: IEEE Conference on Computer Vision and Pattern Recognition (2008)
17. Le, D.D., Zhu, C.-Z., Phan, S., Poullot, S., Duong, D.A., Satoh, S.: National institute of informatics, Japan at trecvid 2013. In: TRECVID, Orlando, Florida, USA (2013)
18. Zhu, C., Jegou, H., Satoh, S.: Query-adaptive asymmetrical dissimilarities for visual object retrieval. In: IEEE International Conference on Computer Vision, ICCV 2013, pp. 1705–1712, Sydney, Australia. IEEE, 1–8 Dec 2013
19. Tolias, G., Avrithis, Y.S.: Speeded-up, relaxed spatial matching. In: IEEE International Conference on Computer Vision, ICCV 2011, pp. 1653–1660. Barcelona, Spain, 6–13 Nov 2011
20. Zhang, W., Ngo, C.-W.: Searching visual instances with topology checking and context modeling. In Proceedings of the 3rd ACM Conference on International Conference on Multimedia Retrieval, ICMR 2013, pp. 57–64. New York, NY, USA (2013)
21. Jegou, H., Douze, M., Schmid, C.: Hamming embedding and weak geometric consistency for large scale image search. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 304–317. Springer, Heidelberg (2008)
22. Cao, Y., Wang, C., Li, Z., Zhang, L., Zhang, L.: Spatial-bag-of-features. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 3352–3359 (2010)
23. Shen, X., Lin, Z., Brandt, J., Avidan, S., Wu, Y.: Object retrieval and localization with spatially-constrained similarity measure and k-nn re-ranking. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 3013–3020 (2012)
24. Elasticsearch. <https://www.elastic.co/products/elasticsearch>. Accessed 10 Sept 2015
25. Apache Lucene. <http://lucene.apache.org/>. Accessed 10 Sept 2015

Defining Membership Functions in Fuzzy Object-Oriented Database Model

Doan Van Thang and Dang Cong Quoc^(✉)

Ho Chi Minh City Industry and Trade College, Ho Chi Minh, Vietnam
{vanthangdn, dangcongquoc1968}@gmail.com

Abstract. In this paper, we focus study the characteristics of fuzzy attributes, object/class, class/superclass basing on approximate semantic approach to hedge algebras (HA). On this basis, we present methods of determining the membership degree on the fuzzy characteristics this.

Keywords: OODB · FOODB · HA

1 Introduction

Fuzzy relational database model, fuzzy object-oriented database model and related problems have been widely researched by many authors in recent years [1–9]. To represent fuzzy information in the data model, there are many basic approaches: the model based on similarity relation and the model based on possibility distribution, etc. All these approaches aim to achieve and handle the fuzzy values to the satisfaction of the incomplete, imprecise and uncertain information.

Depending on advantages of HA structure [4, 5], the authors study a relational database model [6–9] and fuzzy object-oriented [2, 3] based on the approach of HA, in which linguistic semantics is expressed by values of semantically quantifying mappings of HA. According to the approach of HA, linguistic semantics can be expressed in a neighborhood of intervals determined by the fuzzy measure of the linguistic value of an attribute as a linguistic variable.

Similar to traditional object oriented database model, in the fuzzy object oriented database models exist the relationships as relation between the class with objects, between subclasses with superclass, association relationship. The problem is how to determine degree membership the relationships. On that basis, we build degree measure the semantic approximation of the two fuzzy data to define degree membership of relationships this in the fuzzy OODB model.

This paper is presented as follows: Sect. 2 presents some fundamental concepts related to hedge algebraic as the basis for the next section. Section 3 presents the method of determining the degree of membership in the fuzzy OODB model, and Sect. 4 concludes the paper.

2 Hegde Algebra

Consider a complete hedge algebra (Comp-HA) $AX = (X, G, H, \Phi, \Sigma, \leq)$, where G is a set of generators which are designed as primary terms denoted by c^- and c^+ , and specific constants 0 , W and 1 (zero, neutral and unit elements, respectively), $H = H + \cup H^-$ and two artificial hedges Σ , Φ , the meaning of which is, respectively, taking in the poset X the supremum (sup, for short) or infimum (inf, for short) of the set $H(x)$ - the set generated from x by using operations in H . The word “complete” means that certain elements are added to usual hedge algebras in order for the operations Σ and Φ will be defined for all $x \in X$. Set $\text{Lim}(X) = XH(G)$, the set of the so-called limit elements of AX .

Proposition 2.1. Fuzziness measures fm and fuzziness measures of $\mu(h)$, $\forall h \in H$, the following statements hold:

- (1) $fm(hx) = \mu(h)fm(x)$, $\forall x \in X$.
- (2) $fm(c^-) + fm(c^+) = 1$.
- (3) $\sum_{-q \leq i \leq p, i \neq 0} fm(h_i c) = fm(c)$, where $c \in \{c^-, c^+\}$.
- (4) $\sum_{-q \leq i \leq p, i \neq 0} fm(h_i x) = fm(x)$, $x \in X$.
- (5) $\sum \{\mu(h_i) : -q \leq i \leq -1\} = \alpha$ and $\sum \{\mu(h_i) : 1 \leq i \leq p\} = \beta$, where $\alpha, \beta > 0$ và $\alpha + \beta = 1$.

In HA, each term $x \in X$ always have negative sign or positive sign, is called PN-sign and is defined recursively as below:

Definition 2.1. (Sign function). $\text{Sgn}: X \rightarrow \{-1, 0, 1\}$ is a function which is defined recursively as follows, where $h, h' \in H$, and $c \in \{c^-, c^+\}$:

- (1) $\text{Sgn}(c^-) = -1$, $\text{Sgn}(c^+) = +1$.
- (2) $\text{Sgn}(h'hx) = 0$, nếu $h'hx = hx$, otherwise

$\text{Sgn}(h'hx) = -\text{Sgn}(hx)$, if $h'hx \neq hx$ và h' is negative with h (or c , if $h = I$ and $x = c$)
 $\text{Sgn}(h'hx) = +\text{Sgn}(hx)$, if $h'hx \neq hx$ và h' is positive with h (or c , if $h = I$ and $x = c$).

Proposition 2.2. with $\forall x \in X$, we have: $\forall h \in H$, if $\text{Sgn}(hx) = +1$ then $hx > x$, if $\text{Sgn}(hx) = -1$ then $hx < x$ and if $\text{Sgn}(hx) = 0$ then $hx = x$.

From properties of fuzziness and sign function, semantically quantifying mapping of HA is defined as below:

Definition 2.2. Let $AX = (X, G, H, \Sigma, \Phi, \leq)$ be a free linear complete HA, $fm(x)$ and $\mu(h)$ are, respectively, the fuzziness measures of linguistic and the hedge h satisfying properties in proposition 2.1. Then, v is a induced mapping by fuzziness measure fm of the linguistic if it is determined as follows:

- (1) $v(W) = \kappa = fm(c^-)$, $v(c^-) = \kappa - \alpha fm(c^-) = \beta fm(c^-)$, $v(c^+) = \kappa + \alpha fm(c^+)$.

- (2) $v(h_jx) = v(x) + Sgn(h_jx)\{\sum_{i=Sgn(j)}^j \mu(h_i)fm(x) - \omega(h_jx)\mu(h_j)fm(x)\}$, where $\omega(h_jx) = \frac{1}{2}[1 + Sgn(h_jx)Sgn(h_p h_jx)(\beta - \alpha)] \in \{\alpha, \beta\}$, for all j , $-q \leq j \leq p$ and $j \neq 0$
- (3) $v(\Phi c-) = 0, v(\Sigma c-) = \kappa = v(\Phi c+), v(\Sigma c+) = 1$, for all j , $-q \leq j \leq p$ and $j \neq 0$,

We have: $v(h_jx) = v(x) + Sgn(h_jx)\{\sum_{i=Sgn(j)}^{j-1} \mu(h_i)fm(x)\}$ and $v(h_jx) = v(x) + Sgn(h_jx)\{\sum_{i=Sgn(j)}^j \mu(h_i)fm(x)\}$.

Example 1. Let HA AX = (X, C, H, ≤), Where $H^+ = \{\text{More, Very}\}$ with More < Very and $H^- = \{\text{Little, Possibly}\}$ with Little > Possibly. C = {Small, Large} with Small is negative term, Large is positive term. Assuming let W = 0.5, fm(Little) = 0.4, fm(Possibly) = 0.1, fm(More) = 0.1, fm(Very) = 0.4. Meantime, we have Table 1 of values function v as follow.

Table 1. values function v

Linguistic value	Function v	Linguistic value	Function v
Very very small	0.04	Very very large	0.96
Very small	0.10	Very large	0.90
Possibly very small	0.11	Possibly very large	0.89
Little very small	0.16	Little very large	0.84
Small	0.25	Large	0.75
Very possibly small	0.26	Very possibly large	0.74
Little small	0.40	Little large	0.60
More little small	0.41	More little small	0.59
Very little small	0.46	Very little large	0.54

3 Fuzzy Object-Oriented Database Model

FOODB model is proposed data model similarity based. In FOODB model, regarding the representation of imprecise information, uncertainty is handled at three levels: attribute level, class/superclass level and object/class level.

3.1 Attribute Level Uncertainty

3.1.1 Attribute Uncertainty

FOODB deals with 3 types of uncertainty at the attribute level.

- (a) The first type being incomplete type when the value of the attribute is specified as a range value (e.g. 100–200). This type is called “incompleteness.”
- (b) The second type of uncertainty occurs when the value of the attribute is unknown, does not exist or there is no information on whether a value exists or not. This type of uncertainty is called “null”.

- (c) The third type of uncertainty occurs when the value of the attribute is vaguely specified. This type of uncertainty is called “fuzzy”.

A similar relationship or the fuzzy equivalence relationship, is represented by a similarity matrix, is basis for FOODB model the based on similarity. The similarity matrix shows the similarity of each element of other elements in fuzzy domain. The next section, we present an method similar matrix construction based on semantic approximation of HA.

3.1.2 Similarity Matrix

Definition 3.1. To evaluate the semantic approximation of resemblance between each pair term in fuzzy domain of a attribute, we construct function SP (*Semantic Proximity*) as follows:

$$SP(x, y) = 1 - |v(x) - v(y)|$$

where, $v(x)$ and $v(y)$ respectively is quantitative semantics value of the linguistic x and y .

Function SP have the following:

1. $0 \leq SP(x,y) \leq 1$
2. $SP(x,x) = 1$
3. $SP(x,y) = SP(y, x)$

Example 2. Build quantitative semantics for attribute in the case where attribute values are linguistic values. Consider HA of linguistic variable age, where $D_{age} = [0, 100]$, generating elements $\{0, \text{young}, W, \text{old}, 1\}$, the set of hedges are $\{\text{little, possibly, more, very}\}$ (L, P, M, V correspond), $FD_{age} = H_{age}(\text{old}) \cup H_{age}(\text{young})$. Choose $fm(\text{old}) = 0.5$, $fm(\text{young}) = 0.5$, $\mu(P) = 0.2$, $\mu(L) = 0.3$, $\mu(M) = 0.1$ and $\mu(V) = 0.4$.

Based on the definition 2.2 we calculate quantitative values of linguistics term for attribute age, results as follows: $v(V \text{ young}) = 0.1$; $v(M \text{ young}) = 0.225$; $v(\text{young}) = 0.25$; $v(P \text{ young}) = 0.45$; $v(L \text{ young}) = 0.325$; $v(L \text{ old}) = 0.575$; $v(P \text{ old}) = 0.7$; $v(\text{old}) = 0.75$; $v(M \text{ old}) = 0.775$; $v(V \text{ old}) = 0.9$. Since, based on the definition 3.1, we have similarity matrix between each pair of elements in the domain fuzzy of age attribute (Table 2).

The Fuzzy Object-Oriented database model can have multivalued attribute values, and these values may be connected by AND, OR, or XOR semantics. The attributes can have a set of values (leading to multivalued attributes) connected with a logical operator AND/OR/XOR. The attribute value sets are differentiated according to their semantics. The following syntax is used to indicate AND, OR or XOR multivalued attributes: indicate AND is $\langle \dots \rangle$; indicate OR is $\{ \dots \}$; indicate XOR $[\dots]$.

Table 2. Similarity matrix for attribute *age*

Age	V young	M young	Young	P young	L young	L old	P old	Old	M old	V old
V young	1	0.875	0.85	0.65	0.775	0.525	0.4	0.35	0.325	0.2
M young	0.875	1	0.975	0.775	0.9	0.65	0.525	0.475	0.45	0.325
young	0.85	0.975	1	0.8	0.925	0.675	0.55	0.5	0.475	0.35
P young	0.65	0.775	0.8	1	0.875	0.875	0.75	0.7	0.675	0.55
L young	0.775	0.9	0.925	0.875	1	0.75	0.625	0.575	0.55	0.425
L old	0.525	0.65	0.675	0.875	0.75	1	0.875	0.825	0.8	0.675
P old	0.4	0.525	0.55	0.75	0.625	0.875	1	0.95	0.925	0.8
old	0.35	0.475	0.5	0.7	0.575	0.825	0.95	1	0.975	0.85
M old	0.325	0.45	0.475	0.675	0.55	0.8	0.925	0.975	1	0.875
V old	0.2	0.325	0.35	0.55	0.425	0.675	0.8	0.85	0.875	1

3.2 Object/Class Level Uncertainty

Uncertainty at the object/class level refers to the existence of a partial membership of an object to its class. In FOODB model, the boundaries of a class might be uncertain since it has fuzzy attributes. Range of a fuzzy attribute indicates ideal values for that attribute.

Since a fuzzy attribute may take any value from its domain regardless of its range definition, some objects are full members of their classes with a membership degree of 1 whereas some objects are member of their classes with a membership degree changing between 0 and 1. The values of fuzzy attributes of an object determine the membership degree of that object to its class. The closer the value of fuzzy attributes of an object to range definitions, the higher the object membership degree. Relevance of the fuzzy attributes and the similarity between the fuzzy attributes' values and their range definitions determine the membership degree of an object to its class.

Based on the considerations of relevance and inclusion of attribute values, the membership degree of object o_j in class C has been defined as:

$$\mu_C(o_j) = \frac{\sum INC(rng_C(a_i)/o_j(a_i)) * RLV(a_i, C)}{\sum RLV(a_i, C)}$$

Where:

- $INC(rng_C(a_i)/o_j(a_i))$ denotes the degree of inclusion of the attribute values of o_j in the formal range a_i in the class C .
- $RLV(a_i, C)$ indicates the relevance of the attribute a_i to the class C . Degree of inclusion of this represents the relevance of the attribute a_i to the definition of the class C . This degree inclusion is determined based on the concept of affinity attribute by Hoffer and Severance proposed, and cosin measure.

In addition, the weighted average is used to calculate the membership degree of objects. All attributes, therefore, affect the membership degree proportionally to their relevance.

Calculation the degree of inclusion for the different semantics are explained below:

AND semantics: Under AND semantics, an attribute takes more than one value and all values exist simultaneously (are true). AND semantics arise when data is nested, the INC formulation for AND semantics as follows

$$INC(rng_C(a_i)/o_j(a_i)) = Min[Min[Max(SP(x, y))], Min[Max(SP(z, w))]],$$

$$\forall x \in rng_C(a_i), \forall y \in o_j(a_i), \forall z \in o_j(a_i), \forall w \in rng_C(a_i).$$

Note that the basis of comparison is the range definition, comparing all of the elements in the object attribute to each of the entries in the range definition. Then the order is reversed. That is, the attribute definition is the reference point.

OR semantics: Under OR semantics an attribute takes more than one value, all or some of which may exist simultaneously (are true). The original similarity-based model is weak when applying OR and XOR semantics. Reformulation is required since the model treats both the connectives the same way ignoring the semantic difference between them. So here we find the value of to be

$$INC(rng_C(a_i)/o_j(a_i)) = Min[Max(SP(x, z)), Threshold(o_j(a_i))],$$

$$\forall x \in o_j(a_i), \forall z \in rng_C(a_i)$$

In here, the threshold indicates the minimum level of similarity between the elements of the object attribute value, threshold is determined as follows

$$Threshold(o_j(a_i)) = Min[SP(x, z)], \forall x, \forall z \in o_j(a_i)$$

XOR semantics: XOR semantics forces only one of the entries in the tuple be true at a time. We can assume equal probabilities for the entries in the list. The INC formulation for XOR semantics as follows

$$INC(rng_C(a_i)/o_j(a_i)) = Avg[Max(SP(x, y))], \forall x \in o_j(a_i), \forall y \in rng_C(a_i)$$

3.3 Class/Subclass Level Uncertainty

Uncertainty at the class/subclass level refers to the existence of a partial membership of a class to its superclass. This type of uncertainty indicates that the fuzziness occurs at the class inheritance hierarchy since a class hierarchy might not be constructed precisely in some cases. The membership degree of the class C to the class C_i of its and is determined using the formulation:

$$\mu_{C_i}(C) = \frac{\sum INC(rng_{C_i}(a_i)/rng_C(a_i)) * RLV(a_i, C_i)}{\sum RLV(a_i, C_i)}$$

where: $INC = Min(Max[SP(x, y)]), \forall x \in rng_{C_i}(a_i), \forall y \in rng_C(a_i).$

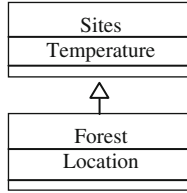


Fig. 1. Inherited relationship for class forest and class sites

Example 3 Consider class hierarchies as follows (Fig. 1).

Build quantitative semantics for attribute in the case where attribute values are linguistic values. Consider HA of linguistic variable Temperature, where $D_{Temperature} = [0, 100]$, generating elements are $\{0, \text{cold}, W, \text{hold}, 1\}$, the set of hedge are $\{\text{little, possibly, more, very}\}$ (L, P, M, V correspond), $FD_{Temperature} = H_{Temperature}(\text{hold}) \cup H_{Temperature}(\text{cold})$. Chosse $fm(\text{hold}) = 0.5$, $fm(\text{cold}) = 0.5$, $\mu(P) = 0.2$, $\mu(L) = 0.3$, $\mu(M) = 0.2$ and $\mu(V) = 0.3$.

Based on the definition 2.2 we can calculate quantitative values of linguistic term for attribute Temperature, results as follows: $\nu(V \text{ cold}) = 0.075$; $\nu(M \text{ cold}) = 0.2$; $\nu(\text{cold}) = 0.25$; $\nu(P \text{ cold}) = 0.3$; $\nu(L \text{ cold}) = 0.425$; $\nu(L \text{ hold}) = 0.575$; $\nu(P \text{ hold}) = 0.7$; $\nu(\text{hold}) = 0.75$; $\nu(M \text{ hold}) = 0.8$; $\nu(V \text{ hold}) = 0.925$.

Based on the definition 3.1, we have been the similarity values: $SP(\text{possibly cold, little hold}) = 0.725$, $SP(\text{possibly cold, hold}) = 0.55$, $SP(\text{hold, little hold}) = 0.825$

We have range values and degree of inclusion for attributes as follows:

$$\begin{aligned} \text{rng}_{\text{Sites}}(\text{Temperature}) &= \{\text{little hold, hold}\} \\ \text{RLV}(\text{Temperature, Sites}) &= 0.5 \\ \text{rng}_{\text{Forest}}(\text{Temperature}) &= \{\text{possibly cold, little hold}\} \end{aligned}$$

We calculate the degree of inclusion for attributes as follows:

$$\text{INC}(\text{rng}_{\text{Sites}}(\text{Temperature})/\text{rng}_{\text{Forest}}(\text{Temperature})) = \text{Min}[\text{Max}(1, 0.725), \text{Max}(0.55, 0.825)] = 0.825$$

Applying formula calculating membership degree of class Forest to the class Sites, we have:

$$\mu_{\text{Sites}}(\text{Forest}) = [\text{INC}(\text{rng}_{\text{Sites}}(\text{Temperature})/\text{rng}_{\text{Forest}}(\text{Temperature})) * \text{RLV}(\text{Temperature, Sites})] / [\text{RLV}(\text{Temperature, Sites})] = (0.825 * 0.5) / 0.5 = 0.825.$$

In some applications we may face a problem of multiple inheritance when multiple superclasses for an object have different values for a field. This problem is not specific to the FOODB model, but in all hierarchical representational systems. In literature it is pointed out that there is no common solution to adequately resolve all cases of multiple inheritance conflicts.

Ambiguity arises when more than one of the superclasses have common attributes and the subclass does not declare explicitly the class from which the attribute was inherited (see Fig. 2).

Clearly the attributes of the ClassD are attrib1, attrib2, attrib3, attrib4, and attrib5, Attrib4, and attrib5 are defined in the class and the others inherited from the superclasses. Ambiguity exists for attrib1 and attrib2. We propose that there is no need to

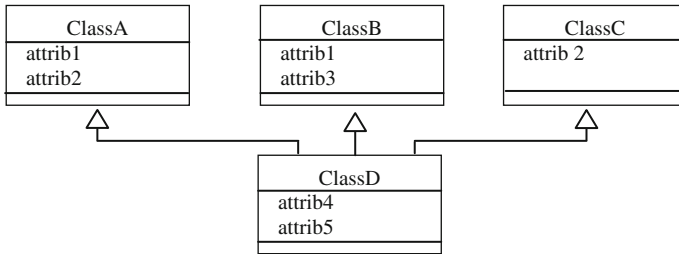


Fig. 2. Ambiguity in multiple inheritance

consider the superclass from which an attribute is not inherited, if it has no such attribute at the class/subclass membership degree calculations.

With this consideration it is possible to reflect changes from the superclass range definitions. Principally, each class is forced to make its range declaration for all its fuzzy attributes. This ensures a declared range definition for a conflicting attribute such as attrib 1 and attrib2 be available in the class definition. The rest is handled by the new formulation for calculating the membership degree of the ClassD to ClassA, ClassB and ClassC. For example, consider the following lattice formed by imaginary chemical substances (Fig. 3). All of the superclasses have the same attribute color. The range definitions are shown. With $rng_{color}(A) = \langle \text{dark_gray} \rangle$, $rng_{color}(B) = \langle \text{yehllow, brown} \rangle$ and $rng_{color}(C) = \langle \text{gray, black} \rangle$.

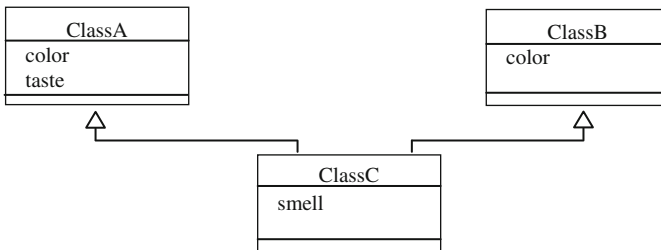


Fig. 3. Lattice of chemical substances

The color attribute of the class ClassC appears to be inherited from ClassA. When applying the INC formula on each superclass to determine the class/subclass membership, this fact is reflected by the results. The inclusion value for the range of ClassC color in ClassA color will result in an INC value of one, as expected. However the results for the ClassB class are much less

$$INC = \min(\max[SP(\text{yellow, gray}), SP(\text{yellow, black})], \max(SP(\text{brown, gray}), SP(\text{brown, black})))$$

showing that the bond between ClassA and ClassC is stronger than ClassB and ClassC for attribute color. Of course, another key point are the relevance rules defined in the classes. If the RLV value for color of ClassA or ClassB is high, the effect of any deviation will be higher.

4 Conclusion

The paper focus on describing the main aspects of the FOODB model the based on the HA. In this model, the fuzzy appeared in three levels: attribute, object/class and class/superclass. In attribute level, there are three types of uncertainty in the value of the attribute. Object/class level refers to the existence of a partial membership of a object to class. Class/superclass level refers to the existence of a partial membership of a class to its. In this paper, we focus on building membership function for the attribute level, object/class and class/superclass and multiple inheritance. On the basis definition the membership function of this, we will present methods the fuzzy query object in the next paper.

References

1. Biazzo, V., Giugno, R., Lukasiewicz, T., Subrahmanian, V.S.: Temporal probabilistic object bases. *IEEE Trans. Knowl. Eng.* **5**, 921–939 (2002)
2. Doan, V.B., Truong, C.T., Doan, V.T.: Querying data with fuzzy information in object-oriented databases based on hedge algebraic semantics. In: *Proceedings of the 4th International Conference on Knowledge and Systems Engineering*, pp 39–45. IEEE Computer Society Press, Da Nang, Vietnam (2012)
3. Doan, V.T.: Dependence fuzzy objects. In *Proceedings of the 2014 International Conference on Advanced Technologies for Communications, Special session on Computational Science and Computational Intelligence*, pp 160–167. IEEE Communications Society, Hanoi, Vietnam (2014)
4. Ho, N.C.: Quantifying hedge algebras and interpolation methods in approximate reasoning. In: *Proceedings of the 5th International Conference on Fuzzy Information Processing*, pp 105–112 (2003)
5. Nguyen, C.H., Wechler, W.: Hedge algebras: an algebraic approach to structures of sets of linguistic domains of linguistic truth variable. *Fuzzy Sets Syst.* **35**(3), 281–293 (1990)
6. Nguyen, C.H.: A topological completion of refined hedge algebras and a model of fuzziness of linguistic terms and hedges. *Fuzzy Sets Syst.* **158**, 436–451 (2007)
7. Nguyen, C.H., Huynh, V.N., Tran, D.K., Le, H.C.: Hedge Algebras, Linguistic- valued logic and their application to fuzzy reasoning. *Int. J. Uncertainty Fuzziness Knowl.-Based Syst.* **7** (4), 347–361 (1999)
8. Cat, H.N., Thai, S.T., Dinh, P.P.: Modeling of a semantics core of linguistic terms based on an extension of hedge algebra semantics and its application. *Knowl.-Based Syst.* **67**, 244–262 (2014)
9. Cat, H.N., Witold, P., Thang, L.D., Thai, S.T.: A genetic design of linguistic terms for fuzzy rule based classifiers. *Int. J. Approximate Reasoning* **54**, 1–20 (2013)

Erratum to: Facilitating the Design/Evaluation Process of Web-Based Geographic Applications: A Case Study with WINDMash

The Nhan Luong^{1(✉)}, Christophe Marquesuzaa², Patrick Etcheverry²,
Thierry Nodenot², and Sébastien Laborie²

¹ Faculty of Computer Science and Engineering,
Ho Chi Minh City University of Technology,
268 Ly Thuong Kiet Street, District 10, Ho Chi Minh City, Vietnam
nhan@hcmut.edu.vn

² Université de Pau et des Pays de l'Adour, Laboratoire d'informatique,
EA 3000, 64600 Anglet, France
{christophe.marquesuzaa,patrick.etccheverry,
thierry.nodenot,sebastien.laborie}@iutbayonne.
univ-pau.fr

Erratum to:

Chapter “Facilitating the Design/Evaluation Process of Web-Based Geographic Applications: A Case Study with WINDMash” in: T.K. Dang et al. (Eds.): Future Data and Security Engineering, LNCS 9446, https://doi.org/10.1007/978-3-319-26135-5_19

In the originally published version of this paper the affiliation of the first author, The Nham Luong, contained a punctuation mistake due to which the city was incorrectly stated as “Chi Minh City” instead of “Ho Chi Minh City”. The affiliation of the authors Christophe Marquesuzaa, Patrick Etcheverry, Thierry Nodenot, and Sébastien Laborie was incorrectly stated as “T2i – LIUPPA, Université de Pau et des Pays de l'Adour, 2 Allée du Parc Montaury, 64600 Anglet, France”. This has been corrected.

The updated online version of this chapter can be found at
https://doi.org/10.1007/978-3-319-26135-5_19

Author Index

- Antunes, Pedro 85, 165
- Binh, Nguyen Thanh 226
- Calvo, Hiram 109
Cao, Thi H. 242
- Dang, Tran Khanh 16
Do, Phung 272
Do, Thanh-Nghi 3, 32
Dung, Tran Nam 272
Duy, Nguyen Huynh Anh 85
- Etcheverry, Patrick 259
- Hai, Nguyen Tri 46
- Jäger, Markus 16
Johnstone, David 85
Jonnavithula, Lalitha 165
- Küng, Josef 16
- Laborie, Sébastien 259
Le, Hong Anh 98
Lê, Lam-Son 183, 211
Le, Son Thanh 57
Le, Tuan Dinh 46
León, Elvia 72
Luong, The Nhan 259
- Marquesuzaà, Christophe 259
Masada, Tomonari 123
Minh, Quang Tran 135
Minh-Thai, Tran Nguyen 147
- Nadschläger, Stefan 16
Nghia, Nguyen Hoang 46
Nguyen, Duc-Hoang 299
Nguyen, Hung 272
Nguyen, Phuong T. 98
Nguyen, Thanh D. 242
Nguyen, Tuan M. 242
Nguyen, Vinh-Tiep 299
Nguyen, Vu Thanh 46, 272
Nodenot, Thierry 259
- Phan, Trong Nhan 16
Poulet, François 3
- Quang-Hung, Nguyen 198
Quoc, Dang Cong 314
- Reyes Daza, Brayan S. 72
- Salcedo Parra, Octavio J. 72
Simões, David 165
- Ta, Chien D.C. 285
Takasu, Atsuhiko 123
Thai-Nghe, Nguyen 147
Thi, Tuoi Phan 285
Thoai, Nam 198
Thuan, Nguyen Hoang 85, 165
Tran, Ha Manh 57
Tran, Minh-Triet 299
- Van Nguyen, Sinh 57
Van Thang, Doan 314
Vu, Quy Tran 57