

Weiru Liu · Fausto Giunchiglia
Bo Yang (Eds.)

LNAI 11061

Knowledge Science, Engineering and Management

11th International Conference, KSEM 2018
Changchun, China, August 17–19, 2018
Proceedings, Part I

1
Part I

 Springer

Lecture Notes in Artificial Intelligence

11061

Subseries of Lecture Notes in Computer Science

LNAI Series Editors

Randy Goebel

University of Alberta, Edmonton, Canada

Yuzuru Tanaka

Hokkaido University, Sapporo, Japan

Wolfgang Wahlster

DFKI and Saarland University, Saarbrücken, Germany

LNAI Founding Series Editor

Joerg Siekmann

DFKI and Saarland University, Saarbrücken, Germany

More information about this series at <http://www.springer.com/series/1244>


Weiru Liu · Fausto Giunchiglia
Bo Yang (Eds.)

Knowledge Science, Engineering and Management

11th International Conference, KSEM 2018
Changchun, China, August 17–19, 2018
Proceedings, Part I

Editors
Weiru Liu
University of Bristol
Bristol
UK

Bo Yang 
Jilin University
Changchun
China

Fausto Giunchiglia 
Università di Trento
Povo
Italy

ISSN 0302-9743 ISSN 1611-3349 (electronic)
Lecture Notes in Artificial Intelligence
ISBN 978-3-319-99364-5 ISBN 978-3-319-99365-2 (eBook)
<https://doi.org/10.1007/978-3-319-99365-2>

Library of Congress Control Number: 2018951241

LNCS Sublibrary: SL7 – Artificial Intelligence

© Springer Nature Switzerland AG 2018

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. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

The International Conference on Knowledge Science, Engineering and Management (KSEM) provides a forum for researchers in the broad areas of knowledge science, knowledge engineering, and knowledge management to exchange ideas and to report state-of-the-art research results. KSEM 2018 was the 11th in this series, building on the success of ten events in Guilin, China (KSEM 2006); Melbourne, Australia (KSEM 2007); Vienna, Austria (KSEM 2009); Belfast, UK (KSEM 2010); Irvine, USA (KSEM 2011); Dalian, China (KSEM 2013); Sibiu, Romania (KSEM 2014); Chongqing, China (KSEM2015); Passau, Germany (KSEM 2016) and Melbourne, Australia (KSEM 2017).

The selection process this year was competitive. We received 262 submissions, and each submitted paper was reviewed by at least three members of the Program Committee (PC). Following this independent review, there were discussions among reviewers and PC chairs. A total of 62 papers were selected as full papers (23.6%), and another 26 as short papers (9.9%), yielding a combined acceptance rate of 33.5%. Moreover, we were honored to have three prestigious scholars giving keynote speeches at the conference: Prof. Anthony Hunter (University College London, UK), Prof. Qiang Yang (The Hong Kong University of Science and Technology, SAR China), and Prof. Meikang Qiu (Pace University, USA). The abstract of Prof. Hunter's talk is included in this volume.

We would like to thank everyone who participated in the development of the KSEM 2018 program. In particular, we want to give special thanks to the PC, for their diligence and concern for the quality of the program, and also for their detailed feedback to the authors. The general organization of the conference also relied on the efforts of the KSEM 2018 Organizing Committee.

Moreover, we would like to express our gratitude to the KSEM Steering Committee honorary chair, Ruqian Lu (Chinese Academy of Sciences, China), as well as the KSEM 2018 general co-chairs, Prof. Qiang Yang (The Hong Kong University of Science and Technology, SAR China), Prof. Joerg Siekmann (DFKI and Saarland University, Germany), and Prof. Xiaohui Wei (Jilin University, China). We are also grateful to the team at Springer led by Alfred Hofmann for publication of this volume.

Finally, and most importantly, we thank all the authors, who are the primary reason that KSEM 2018 was so exciting and why it remains the premier forum for presentation and discussion of innovative ideas, research results, and experience from around the world.

June 2018

Weiru Liu
Fausto Giunchiglia
Bo Yang

Organization

KSEM Steering Committee

Steering Committee

Ruqian Lu (Honorary Chair)	Chinese Academy of Sciences, China
Chengqi Zhang (Past Chair)	University of Technology Sydney, Australia
Hui Xiong (Chair)	The State University of New Jersey, USA
Dimitris Karagiannis (Deputy Chair)	University of Vienna, Austria
David Bell	Queen's University Belfast, UK
Yaxin Bi	Ulster University, UK
Cungen Cao	Chinese Academy of Sciences, China
Zhi Jin	Peking University, China
Claudiu Kifor	Lucian Blaga University of Sibiu, Romania
Jérôme Lang	University Paul Sabatier, France
Yoshiteru Nakamori	JAIST, Japan
Joerg Siekmann	DFKI and Saarland University, Germany
Eric Tsui	The Hong Kong Polytechnic University, SAR China
Zongtuo Wang	Dalian Science and Technology University, China
Kwok Kee Wei	City University of Hong Kong, SAR China
Martin Wirsing	Ludwig-Maximilians-Universität München, Germany
Mingsheng Ying	Tsinghua University, China
Zili Zhang	Southwest University, China

KSEM 2018 Organizing Committee

General Co-chairs

Qiang Yang	The Hong Kong University of Science and Technology, SAR China
Joerg Siekmann	DFKI and Saarland University, Germany
Xiaohui Wei	Jilin University, China

Program Committee Co-chairs

Weiru Liu	University of Bristol, UK
Fausto Giunchiglia	University of Trento, Italy
Bo Yang	Jilin University, China

Publicity Co-chairs

Françoise Fogelman-Soulié Tianjin University, China
 Xiangjiu Che Jilin University, China

Organization Co-chairs

Jihong Ouyang Jilin University, China
 Daxin Zhang Jilin University, China

KSEM 2018 Program Committee

Andreas Albrecht	Middlesex University, UK
Klaus-Dieter Althoff	DFKI/University of Hildesheim, Germany
Serge Autexier	DFKI, Germany
Costin Badica	University of Craiova, Romania
Salem Benferhat	Université d'Artois, France
Philippe Besnard	CNRS/IRIT, France
Remus Brad	Lucian Blaga University of Sibiu, Romania
Robert Andrei Buchmann	Babeş-Bolyai University, Romania
Paolo Ciancarini	University of Bologna, Italy
Ireneusz Czarnowski	Gdynia Maritime University, Poland
Richard Dapoigny	LISTIC/Polytech'Savoie, France
Yong Deng	Southwest University, China
Josep Domingo-Ferrer	Universitat Rovira i Virgili, Spain
Susan Elias	VIT University Chennai Campus, India
Dieter Fensel	University of Innsbruck, Austria
Hans-Georg Fill	University of Bamberg, Germany
Yanjie Fu	Missouri University of Science and Technology, USA
Vijayabharadwaj Gsr	VIT University Chennai Campus, India
Knut Hinkelmann	FHNW University of Applied Sciences and Arts Northwestern Switzerland, Switzerland
Zhisheng Huang	Vrije University Amsterdam, The Netherlands
Van Nam Huynh	JAIST, Japan
Zhi Jin	Peking University, China
Fang Jin	Texas Tech University, USA
Mouna Kamel	IRIT – Université Paul Sabatier – Toulouse, France
Konstantinos Kotis	University of Piraeus, Greece
Li Li	Southwest University, China
Huayu Li	The University of North Carolina at Charlotte, USA
Gang Li	Deakin University, Australia
Ge Li	Peking University, China
Qian Li	Chinese Academy of Sciences, China
Junming Liu	Rutgers University, USA
Li Liu	Chongqing University, China
Shaowu Liu	University of Technology Sydney, Australia
Bin Liu	IBM Thomas J. Watson Research Center, USA

Xudong Luo	Guangxi Normal University, China
Bo Ma	Chinese Academy of Sciences, China
Stewart Massie	Robert Gordon University, UK
Maheswari N.	VIT University, Chennai, India
Oleg Okun	Cognizant Technology Solutions GmbH, Germany
Dan Oleary	University of Southern California, USA
Dantong Ouyang	Jilin University, China
Maurice Pagnucco	The University of New South Wales, Australia
Tulasi Prasad	VIT University, India
Guilin Qi	Southeast University, China
Sven-Volker Rehm	WHU – Otto Beisheim School of Management, Germany
Ulrich Reimer	University of Applied Sciences St. Gallen, Switzerland
Gheorghe Cosmin Silaghi	Babeş-Bolyai University, Romania
Leilei Sun	Dalian University of Technology, China
Jianlong Tan	Chinese Academy of Sciences, China
Lucian Vintan	Lucian Blaga University of Sibiu, Romania
Daniel Volovici	Lucian Blaga University of Sibiu, Romania
Huy Quan Vu	Victoria University, Australia
Kewen Wang	Griffith University, Australia
Hongtao Wang	Chinese Academy of Science, China
Zhichao Wang	Tsinghua University, China
Martin Wirsing	Ludwig-Maximilians-Universität München, Germany
Robert Woitsch	BOC Asset Management, Austria
Zhiang Wu	Nanjing University of Finance and Economics, China
Le Wu	Hefei University of Technology, China
Tong Xu	University of Science and Technology of China, China
Ziqi Yan	Beijing Jiaotong University, China
Jingyuan Yang	Rutgers University, USA
Feng Yi	Chinese Academy of Sciences, China
Qingtian Zeng	Shandong University of Science and Technology, China
Chunxia Zhang	Beijing Institute of Technology, China
Le Zhang	Sichuan University, China
Songmao Zhang	Chinese Academy of Sciences, China
Zili Zhang	Southwest University, China
Hongke Zhao	University of Science and Technology of China, China
Jiali Zuo	Jiangxi Normal University, China

A Brief Introduction to Probabilistic Argumentation (Invited Talk)

Anthony Hunter

Department of Computer Science, University College London, London, UK

Abstract. Argumentation can be modelled at an abstract level using an argument graph (i.e. a directed graph where each node denotes an argument and each arc denotes an attack by one argument on another). Since argumentation involves uncertainty, it is potentially valuable to consider how this can be quantified in argument graphs. Two key approaches to capturing uncertainty in argumentation are the constellations approach and the epistemic approach. The former can be used to represent uncertainty over the topology of the argument graph, and the latter can be used to represent belief in arguments. Theoretical foundations, and studies with participants, are being developed for both approaches.

Keywords: Abstract argumentation · Probabilistic argumentation
Epistemic argumentation

1 Introduction

Abstract argumentation provides an elegant formalism for representing arguments and counterarguments in the form of a directed graph [2]. Each node in the graph denotes an argument, and each arc denotes an attack (i.e. for an arc from A to B , A attacks B , or equivalently A is a counterargument to B). Abstract argumentation also provides criteria for determining acceptable subsets of arguments in the graph by dialectical criteria.

Uncertainty can arise in argumentation for various reasons. For an individual argument, there can be uncertainty in the premises of the argument, or in whether the claim follows from the premises. Many arguments are enthymemes, which means that an argument might not explicitly present its premises and/or claim. This means that when an argument is heard or read by someone, that recipient has to decode the enthymeme to recover the intended argument. There is therefore uncertainty in this decoding process as the decoded argument could be different to the intended argument. So using enthymemes introduces uncertainty as to the content of individual arguments, and uncertainty as to whether one argument attacks another argument.

Further uncertainty arises in dialogical argumentation. When one agent is presenting an argument to another agent, the agent presenting the argument is unsure what related arguments the intended recipient is aware of, and what beliefs the intended recipient might have concerning those arguments. For the agent presenting the argument, this might be important if for example, the agent wants to persuade the other agent to accept a specific argument. In this kind of scenario, the persuader has to make good choices of argument based on what arguments and attacks she thinks the persuadee is aware of and believes.

Probabilistic approaches for modeling uncertainty in argumentation include the constellations approach and the epistemic approach [4]. The first is based on a probability distribution over the subgraphs of the argument graph ([3] which extends [1] and [7]), and this can be used to represent the uncertainty over the structure of the graph (i.e. whether a particular argument or attack appears in the argument graph under consideration). The second approach is the epistemic approach which involves a probability distribution over the subsets of the arguments [4, 6, 10]. This can be used to represent the uncertainty over which arguments are believed. The epistemic approach can be constrained (using axioms or postulates) to be consistent with Dung's dialectical semantics, but it can also be used as a potential valuable alternative to Dung's dialectical semantics. The epistemic approach has been extended to also allow a probability distribution over subsets of attacks, and thereby represent belief in each attack [9].

In a study on dialogical argumentation, participants were asked to express belief in individual arguments using a 7 point Likert scale, and categorize relationships between arguments as one of attack, support, dependent somehow, or unrelated [8]. The results from the study support the use of the constellations approach since people may interpret statements and relations between them differently. The results also support the use of the epistemic approach: (1) people may assign levels of agreements to statements going beyond the 3-valued Dung's approach; (2) the epistemic postulates, in contrast to the classical Dung semantics, can be highly adhered to; and (3) the extended epistemic postulates allow us to model situations where strength of argument and strength of attack are decoupled.

So the theoretical basis of probabilistic approaches, and grounding in studies with participants, are being established. Furthermore, applications are emerging (e.g. computational persuasion [5]).

References

1. Dung, P., Thang, P.: Towards (probabilistic) argumentation for jury-based dispute resolution. In: Proceedings of COMMA'10. FAIA, vol. 216, pp. 171–182. IOS Press (2010)
2. Dung, P.M.: On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artif. Intell.* **77**(2), 321–358 (1995)
3. Hunter, A.: Some foundations for probabilistic abstract argumentation. In: Proceedings of COMMA'12. FAIA, vol. 245, pp. 117–128. IOS Press (2012)

4. Hunter, A.: A probabilistic approach to modelling uncertain logical arguments. *Int. J. Approx. Reason.* **54**(1), 47–81 (2013)
5. Hunter, A.: Towards a framework for computational persuasion with applications in behaviour change. *Argument Comput.* **9**(1), 15–40 (2018)
6. Hunter, A., Thimm, M.: Probabilistic reasoning with abstract argumentation frameworks. *J. Artif. Intell. Res.* **59**, 565–611 (2017)
7. Li, H., Oren, N., Norman, T.J.: Probabilistic argumentation frameworks. In: Modgil, S., et al. (eds.) TAFAs 2011. LNCS, vol. 7132, pp. 1–16. Springer, Berlin (2011)
8. Polberg, S., Hunter, A.: Empirical evaluation of abstract argumentation: Supporting the need for bipolar and probabilistic approaches. *Int. J. Approx. Reason.* **93**, 487–543 (2018)
9. Polberg, S., Hunter, A., Thimm, M.: Belief in attacks in epistemic probabilistic argumentation. In: Moral, S., et al. (eds.) SUM 2017. LNCS, vol. 10564, pp. 223–236. Springer, Cham (2017)
10. Thimm, M.: A probabilistic semantics for abstract argumentation. In: Proceedings of ECAI 2012. FAIA, vol. 242, pp. 750–755. IOS Press (2012)

Contents – Part I

Text Mining and Document Analysis

Sentence Compression with Reinforcement Learning	3
<i>Liangguo Wang, Jing Jiang, and Lejian Liao</i>	
A Biomedical Question Answering System Based on SNOMED-CT	16
<i>Xinhua Zhu, Xuechen Yang, and Hongchao Chen</i>	
Authorship Attribution for Short Texts with Author-Document Topic Model.	29
<i>Haowen Zhang, Peng Nie, Yanlong Wen, and Xiaojie Yuan</i>	
WalkToTopics: Inferring Topic Relations from a Feature Learning Perspective	42
<i>Linan Gao, Zeyu Wang, and Shanqing Guo</i>	
Distant Domain Adaptation for Text Classification	55
<i>Zhenlong Zhu, Yuhua Li, Ruixuan Li, and Xiwu Gu</i>	
Attention Aware Bidirectional Gated Recurrent Unit Based Framework for Sentiment Analysis.	67
<i>Zhengxi Tian, Wenge Rong, Libin Shi, Jingshuang Liu, and Zhang Xiong</i>	
Neural Sentiment Classification with Social Feedback Signals	79
<i>Tao Wang, Yuanxin Ouyang, Wenge Rong, and Zhang Xiong</i>	
A Concept for Generating Business Process Models from Natural Language Description	91
<i>Krzysztof Honkisz, Krzysztof Kluza, and Piotr Wiśniewski</i>	
A Study on Performance Sensitivity to Data Sparsity for Automated Essay Scoring	104
<i>Yanhua Ran, Ben He, and Jungang Xu</i>	
Extract Knowledge from Web Pages in a Specific Domain.	117
<i>Yihong Lu, Shuiyuan Yu, Minyong Shi, and Chunfang Li</i>	
TCMEF: A TCM Entity Filter Using Less Text	125
<i>Hualong Zhang, Shuzhi Cheng, Liting Liu, and Wenxuan Shi</i>	

Image and Video Data Analysis

Two-Stage Object Detection Based on Deep Pruning
for Remote Sensing Image 137
Shengsheng Wang, Meng Wang, Xin Zhao, and Dong Liu

W-Shaped Selection for Light Field Super-Resolution 148
*Bing Su, Hao Sheng, Shuo Zhang, Da Yang, Nengcheng Chen,
and Wei Ke*

Users Personalized Sketch-Based Image Retrieval
Using Deep Transfer Learning 160
Qiming Huo, Jingyu Wang, Qi Qi, Haifeng Sun, Ce Ge, and Yu Zhao

Enhancing Network Flow for Multi-target Tracking
with Detection Group Analysis 169
*Chao Li, Kun Qian, Jiahui Chen, Guangtao Xue, Hao Sheng,
and Wei Ke*

Combine Coarse and Fine Cues: Multi-grained Fusion Network
for Video-Based Person Re-identification 177
Chao Li, Lei Liu, Kai Lv, Hao Sheng, and Wei Ke

Data Processing and Data Mining

Understand and Assess People’s Procrastination by Mining
Computer Usage Log. 187
*Ming He, Yan Chen, Qi Liu, Yong Ge, Enhong Chen, Guiquan Liu,
Lichao Liu, and Xin Li*

Group Outlying Aspects Mining 200
Shaoni Wang, Haiyang Xia, Gang Li, and Jianlong Tan

Fine-Grained Correlation Learning with Stacked Co-attention Networks
for Cross-Modal Information Retrieval. 213
*Yuhang Lu, Jing Yu, Yanbing Liu, Jianlong Tan, Li Guo,
and Weifeng Zhang*

Supervised Manifold-Preserving Graph Reduction
for Noisy Data Classification 226
Zhiqiang Xu and Li Zhang

Personalize Review Selection Using PeRView 238
*Muhmmad Al-khiza’ay, Noora Alallaq, Qusay Alanoz, Adil Al-Azzawi,
and N. Maheswari*

An Online GPS Trajectory Data Compression Method
Based on Motion State Change 250
Hui Wang, Shuang Liu, and Chengcheng Qian

Mining Temporal Discriminant Frames via Joint Matrix Factorization:
A Case Study of Illegal Immigration in the U.S. News Media 260
Qingchun Bai, Kai Wei, Mengwei Chen, Qinmin Hu, and Liang He

Enhancing Cluster Center Identification in Density Peak Clustering. 268
Jian Hou, Aihua Zhang, Chengcong Lv, and Xu E

An Improved Weighted ELM with Hierarchical Feature Representation
for Imbalanced Biomedical Datasets 276
*Liyuan Zhang, Jiashi Zhao, Huamin Yang, Zhengang Jiang,
and Weili Shi*

Recommendation Algorithms and Systems

SERL: Semantic-Path Biased Representation Learning
of Heterogeneous Information Network 287
*Haining Tan, Weiqiang Tang, Xinxin Fan, Quanliang Jing,
and Jingping Bi*

Social Bayesian Personal Ranking for Missing Data in Implicit
Feedback Recommendation 299
Yijia Zhang, Wanli Zuo, Zhenkun Shi, Lin Yue, and Shining Liang

A Semantic Path-Based Similarity Measure for Weighted
Heterogeneous Information Networks 311
*Chunxue Yang, Chenfei Zhao, Hengliang Wang, Riming Qiu, Yuan Li,
and Kedian Mu*

Cross-Domain Recommendation for Mapping Sentiment Review Pattern 324
Yang Xu, Zhaohui Peng, Yupeng Hu, Xiaoguang Hong, and Wenjing Fu

Fuzzy Gravitational Search Approach to a Hybrid Data Model
Based Recommender System 337
Shruti Tomer, Sushama Nagpal, Simran Kaur Bindra, and Vipra Goel

Probabilistic Models and Applications

Causal Discovery with Bayesian Networks Inductive Transfer 351
Haiyang Jia, Zuoxi Wu, Juan Chen, Bingguang Chen, and Sicheng Yao

Robust Detection of Communities with Multi-semantics
in Large Attributed Networks 362
*Di Jin, Ziyang Liu, Dongxiao He, Bogdan Gabrys,
and Katarzyna Musial*

Dual Sum-Product Networks Autoencoding 377
Shengsheng Wang, Hang Zhang, Jiayun Liu, and Qiang-yuan Yu

Recognizing Diseases from Physiological Time Series Data
Using Probabilistic Model 388
Danni Wang, Li Liu, Guoxin Su, Yande Li, and Aamir Khan

Knowledge Engineering Applications

An Incremental Approach Based on the Coalition Formation Game Theory
for Identifying Communities in Dynamic Social Networks 403
Qing Xiao, Peizhong Yang, Lihua Zhou, and Lizhen Wang

LogRank: An Approach to Sample Business Process Event Log
for Efficient Discovery. 415
Cong Liu, Yulong Pei, Qingtian Zeng, and Hua Duan

Case-Based Decision Support System with Contextual Bandits Learning
for Similarity Retrieval Model Selection. 426
Booma Devi Sekar and Hui Wang

Cross-Layer Attack Path Exploration for Smart Grid Based on Knowledge
of Target Network. 433
WenJie Kang, PeiDong Zhu, Gang Hu, Zhi Hang, and Xin Liu

Exploring Cyber-Security Issues in Vessel Traffic Services 442
*Eleni Maria Kalogeraki, Spyridon Papastergiou, Nineta Polemi,
Christos Douligeris, and Themis Panayiotopoulos*

Prognosis of Thyroid Disease Using MS-Apriori Improved Decision Tree . . . 452
*Yuwei Hao, Wanli Zuo, Zhenkun Shi, Lin Yue, Shuai Xue,
and Fengling He*

Stock Price Prediction Using Time Convolution Long
Short-Term Memory Network. 461
*Xukuan Zhan, Yuhua Li, Ruixuan Li, Xiwu Gu, Olivier Habimana,
and Haozhao Wang*

Web Data Extraction from Scientific Publishers’ Website
Using Hidden Markov Model 469
Jing Huang, Ziyu Liu, Beibei Wang, Mingyue Duan, and Bo Yang

Knowledge Graph and Knowledge Management

MedSim: A Novel Semantic Similarity Measure in Bio-medical Knowledge Graphs 479
Kai Lei, Kaiqi Yuan, Qiang Zhang, and Ying Shen

A Sequence Transformation Model for Chinese Named Entity Recognition 491
Qingyue Wang, Yanjing Song, Hao Liu, Yanan Cao, Yanbing Liu, and Li Guo

An Incremental Reasoning Algorithm for Large Scale Knowledge Graph 503
Yifei Wang and Jie Luo

Relation Classification Using Coarse and Fine-Grained Networks with SDP Supervised Key Words Selection. 514
Yiping Sun, Yu Cui, Jinglu Hu, and Weijia Jia

Author Index 523

Contents – Part II

Constraints and Satisfiability

A Multi-objective Optimization Algorithm Based on Preference Three-Way Decomposition	3
<i>Zhao Fu, Hong Yu, Hongliang Zhang, and Xiaofang Chen</i>	
A Community-Division Based Algorithm for Finding Relations Among Linear Constraints	12
<i>Minghao Liu, Feifei Ma, and Jun Yan</i>	
The New Adaptive ETLBO Algorithms with K-Armed Bandit Model	24
<i>Xitong Wang, Yonggang Zhang, and Jiayu Cui</i>	
Enhancing Bug Report Assignment with an Optimized Reduction of Training Set	36
<i>Miaomiao Wei, Shikai Guo, Rong Chen, and Jian Gao</i>	
An Efficient Approach for Computing Conflict Sets Combining Failure Probability with SAT.	48
<i>Ya Tao, Dantong Ouyang, Meng Liu, and Liming Zhang</i>	
A New Variable-Oriented Propagation Scheme for Constraint Satisfaction Problem	59
<i>Zhe Li, Mingqi Yang, and Zhanshan Li</i>	
A Timeline Representation for the Jade Rabbit Rover	69
<i>Dunbo Cai, Yuhui Gao, Wei Gao, and Minghao Yin</i>	
A Parthenogenetic Algorithm for Deploying the Roadside Units in Vehicle Networks	78
<i>Jingli Wu, Yong Wu, Jinyan Wang, and Yutong Ye</i>	

Formal Reasoning and Ontologies

Another Useful Four-Valued Logic	89
<i>Zuoquan Lin and Zhaocong Jia</i>	
An Improved Multi-agent Epistemic Planner via Higher-Order Belief Change Based on Heuristic Search	102
<i>Zhongbin Wu</i>	

ROSIE: Runtime Optimization of SPARQL Queries over RDF
Using Incremental Evaluation 117
Lei Gai, Xiaoming Wang, and Tengjiao Wang

Automated Reasoning over Provenance-Aware Communication Network
Knowledge in Support of Cyber-Situational Awareness 132
*Leslie F. Sikos, Markus Stumptner, Wolfgang Mayer,
Catherine Howard, Shaun Voigt, and Dean Philp*

Constructive Justification Extraction for OWL Ontologies 144
Yuxin Ye, Ling Zhang, Dantong Ouyang, and Mengyu Gao

Deep Learning

A Hybrid RNN-CNN Encoder for Neural Conversation Model 159
Zhiyuan Ma, Wenge Rong, Yanmeng Wang, Libin Shi, and Zhang Xiong

Cross-Dataset Person Re-identification Using Similarity Preserved
Generative Adversarial Networks 171
Jianming Lv and Xintong Wang

Autoencoder Based Community Detection with Adaptive Integration
of Network Topology and Node Contents 184
Jinxin Cao, Di Jin, and Jianwu Dang

Deep Convolutional Nets for Pulmonary Nodule
Detection and Classification 197
Nannan Sun, Dongbao Yang, Shancheng Fang, and Hongtao Xie

Recognizing Character-Matching CAPTCHA Using Convolutional Neural
Networks with Triple Loss 209
Junfeng Hu, Wenchao Ma, Aamir Khan, and Li Liu

Sentiment Embedded Semantic Space for More Accurate
Sentiment Analysis 221
*Jianguo Jiang, Yue Lu, Min Yu, Gang Li, Chao Liu,
Weiqing Huang, and Fangtao Zhang*

Citation Classification Using Multitask Convolutional Neural
Network Model. 232
Abdallah Yousif, Zhendong Niu, and Ally S. Nyamawe

P-DBL: A Deep Traffic Flow Prediction Architecture
Based on Trajectory Data. 244
Jingyuan Wang, Xiaofei Xu, Jun He, and Li Li

Video Restoration Using Convolutional Neural Networks for Low-Level FPGAs	255
<i>Kwok-Wai Hung, Chaoming Qiu, and Jianmin Jiang</i>	
Research on Distribution Alignment and Semantic Consistency in the Adversarial Domain Adaptation	266
<i>Jingcheng Ni, Haiyang Jia, Fangyuan Zhang, Yixuan Wang, and Juan Chen</i>	
Identification of Seismic Wave First Arrivals from Earthquake Records via Deep Learning	274
<i>Yang Yu, Jianfeng Lin, Lei Zhang, Guiquan Liu, Jing Hu, Yuyang Tan, and Haijiang Zhang</i>	
A Deep Network Based on Multiscale Spectral-Spatial Fusion for Hyperspectral Classification.	283
<i>Zhaokui Li, Lin Huang, Deyuan Zhang, Cuiwei Liu, Yan Wang, and Xiangbin Shi</i>	
Symmetric Rectified Linear Units for Fully Connected Deep Models.	291
<i>He Hu</i>	
Network Knowledge Representation and Learning	
Online Kernel Selection with Multiple Bandit Feedbacks in Random Feature Space.	301
<i>Junfan Li and Shizhong Liao</i>	
An Overlapping Microblog Community Detection Method Using New Partition Criterion.	313
<i>Huifang Ma, Meng Xie, Jiahui Wei, and Tingnian He</i>	
Quantifying the Emergence of New Domains: Using Cybersecurity as a Case	324
<i>Xiaoli Hu, Zhiyong Feng, Shizhan Chen, Dongxiao He, and Keman Huang</i>	
Improved Sublinear Primal-Dual Algorithm for Support Vector Machines . . .	337
<i>Ming Gu and Shizhong Liao</i>	
Research on Hot Micro-blog Forecast Based on XGBOOST and Random Forest	350
<i>Jianrong Wang, Chao Lou, Ruiguo Yu, Jie Gao, Tianyi Xu, Mei Yu, and Haibo Di</i>	

Trust-Distrust Aware Recommendation by Integrating Metric Learning with Matrix Factorization	361
<i>Xianglin Zuo, Xing Wei, and Bo Yang</i>	
Joint Detection of Topic Entity and Relation for Simple Question Answering	371
<i>Yunqi Qiu, Yuanzhuo Wang, and Xiaolong Jin</i>	
A Network Embedding-Enhanced Approach for Generalized Community Detection	383
<i>Dongxiao He, Xue Yang, Zhiyong Feng, Shizhan Chen, and Françoise Fogelman-Soulié</i>	
Block Modelling and Learning for Structure Analysis of Networks with Positive and Negative Links	396
<i>Xuehua Zhao, Hua Chen, Xueyan Liu, Xu Tan, and Wenzhuo Song</i>	
An Algorithm of Influence Maximization in Social Network Based on Local Structure Characteristics	403
<i>Yong Wang, Bohan Zhang, Jiahao Shi, Jing Yang, and Jianpei Zhang</i>	
Graphical Models with Content Relevance for Crucial Date Detection in Social Media Event	413
<i>Ruifang He and Dongtai Ding</i>	
Social Knowledge Analysis and Management	
Partially Observable Reinforcement Learning for Sustainable Active Surveillance	425
<i>Hechang Chen, Bo Yang, and Jiming Liu</i>	
Measuring the Diversity and Dynamics of Mobility Patterns Using Smart Card Data	438
<i>Chengmei Liu, Chao Gao, and Yingchu Xin</i>	
Traffic Flow Fluctuation Analysis Based on Beijing Taxi GPS Data	452
<i>Jingyi Guo, Xianghua Li, Zili Zhang, and Junwei Zhang</i>	
Topic Extraction of Events on Social Media Using Reinforced Knowledge. . .	465
<i>Xuefei Zhang and Ruifang He</i>	
APS-PBW: The Analysis and Prediction System of Customer Flow Data Based on WIFI Probes.	477
<i>Yuanyuan Wu, Shunhua Gu, Tong Yu, and Xiaolong Xu</i>	
Author Index	489

Text Mining and Document Analysis



Sentence Compression with Reinforcement Learning

Liangguo Wang¹, Jing Jiang², and Lejian Liao¹(✉)

¹ Beijing Institute of Technology, Beijing, China
{chenwangliangguo, liaolj}@bit.edu.cn

² Singapore Management University, Singapore, Singapore
jingjiang@smu.edu.sg

Abstract. Deletion-based sentence compression is frequently formulated as a constrained optimization problem and solved by integer linear programming (ILP). However, ILP methods searching the best compression given the space of all possible compressions would be intractable when dealing with overly long sentences and too many constraints. Moreover, the hard constraints of ILP would restrict the available solutions. This problem could be even more severe considering parsing errors. As an alternative solution, we formulate this task in a reinforcement learning framework, where hard constraints are used as rewards in a soft manner. The experiment results show that our method achieves competitive performance with a large improvement on the speed.

Keywords: Sentence compression · Deep reinforcement learning

1 Introduction

Sentence compression aims to compress long, verbose sentences into short, concise ones that are grammatical and retain the most important pieces of information. Many applications can benefit from sentence compression. For example, it can be used as a component of a text summarization system [1, 7] or to generate short text which is more suitable for small screens like mobile phones [19].

The task is often expressed as a word deletion problem: Given a sentence containing a sequence of words, the task can be reduced to deleting a subset of these words [9]. Later work generally followed this setting [3, 11], but most of the work assumes a supervised setting, utilizing parallel data to learn mappings between source and compressed sentences [9, 11]. However, parallel labeled data is not always easy to obtain. So in this paper we focus on sentence compression in an unsupervised setting.

When the task is solved in an unsupervised manner, previous work typically formulates it as a constrained optimization problem [1, 3]. Then Integer Linear Programming (ILP) is often used to search for the global optimal solution. Although ILP is guaranteed to find a global optimal solution, this kind of methods faces two problems: (1) ILP is non-deterministic polynomial-time hard

(NP-hard), so its complexity would prevent it from being used in some real-time applications. For example, in our experiments, when we compressed a long sentence with 100 words, it took the ILP method 81 s to finish. When the sentence length grew to 120 words, it took the ILP method 423 s. (2) The constraints of ILP method are hard constraints, which would in some cases prevent valid compression from being considered.

In this study, our target is to design a new unsupervised sentence compression model which can achieve comparable performance compared with ILP-based methods but requires less computation time. A secondary goal is to remove the hard constraints set by ILP method. Specifically, we choose to use deep reinforcement learning in this study motivated by [16]. We formulate sentence compression as a Markov decision process, then a special pointer network is designed to learn the action-value function based on the Deep Q Network (DQN) framework. Besides, hard constraints in ILP method are expressed as soft rewards in our DQN. In this way, we make a compromise between grammaticality and expressive power.

We evaluate our method using various publicly available datasets. In terms of performance, our method achieves comparable result with that of an ILP method based on accuracy and ROUGE scores. Regarding speed, our method works much faster than ILP method.

2 Related Work

2.1 Unsupervised Sentence Compression

Since our research does not use any parallel labeled dataset, we review unsupervised methods only. Compared to supervised methods on this task, there has been less work exploring unsupervised methods. The early work could date back to [6]. They proposed a model to automatically compress text by scoring candidate compressions using a language model combined with a significance score and a score representing the speech recognizer’s confidence in transcribing a given word correctly. [18] used rules to generate compressions. After that, [1] formulated the problem as a constrained optimization problem. They proposed three models, where the first model was totally unsupervised which optimizes a language model based score with some length constraints. [3] hypothesized that syntactic features were more suitable for this task so they used an ILP framework to prune dependency trees.

Compared with these existing studies, to the best of our knowledge, we are the first to apply reinforcement learning for sentence compression by maximizing an objective function.

2.2 Reinforcement Learning in NLP

Deep reinforcement learning has gained attention in recent years and been applied to many natural language processing (NLP) tasks, including information extraction [13], machine translation [5] and summarization [14]. [16] used

reinforcement learning to text summarization and achieved comparable performance with much less time consumed. The main differences between their model and ours are as follows: (1) Their task is sentence selection based summarization while ours is about sentence compression. (2) Our method is deep Q learning network while theirs are traditional Q learning. (3) About the framework, they optimize an objective function that adds sentences to a summary, while we aim to learn to delete tokens from a sentence. We also have completely different objective functions.

3 Method

In this part, we first formulate this task and then introduce the ILP method proposed by [1]. After that, we propose our deep reinforcement learning method.

3.1 Problem Definition

We focus on a deletion-based sentence compression problem, which is the same as that defined by [9]. Let us use $\mathbf{s} = (w_1, w_2, \dots, w_n)$ to denote an input sentence where w_i is a word in the vocabulary V . We would like to delete some of the words in \mathbf{s} to obtain a compressed sentence that still contains the most important information in \mathbf{s} . The compressed sentence can be represented by a sequence of binary labels $\mathbf{y} = (y_1, y_2, \dots, y_n)$, where $y_i = 0$ indicates that w_i is deleted and $y_i = 1$ means w_i is retained.

We assume we do not have any labeled data. Instead, we have only a set of unlabeled data denoted as $\mathcal{D} = s^N$. For any unseen sentence, our goal is to generate a label sequence \mathbf{y} based on an unsupervised model.

3.2 ILP Method

In this section, we review the unsupervised method proposed by [1]. Note that they described three different kinds of methods: unsupervised, semi-supervised and supervised. We focus on the unsupervised method only according to our setting.

The unsupervised method by [1] turns the sentence compression problem into an optimization problem, where the objective is to maximize a scoring function that measures how likely a language model can generate the compressed sentence. Specifically, [1] used a trigram language model as shown in the objective function below:

$$\begin{aligned}
 \max \sum_{i=1}^n \alpha_i \cdot p(x_i | start) \\
 + \sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} \sum_{k=j+1}^n \gamma_{ijk} \cdot p(x_k | x_i, x_j) \\
 + \sum_{i=0}^{n-1} \sum_{j=i+1}^n \beta_{ij} \cdot p(end | x_i, x_j),
 \end{aligned} \tag{1}$$

where x_i is the i -th word in the sentence, α_i is a binary variable indicating whether x_i starts the compression, β_{ij} indicates whether the sequence x_i, x_j ends the compression and γ_{ijk} shows whether the sequence x_i, x_j, x_k is inside the compressed sentence. This objective function allows any combination of trigrams to be selected, including some invalid combinations of trigram sequences such as having two or more trigrams containing the *end* token. Therefore, several sequential constraints were introduced to restrict the set of allowed trigram combinations. E.g., a sentence could only start by one word, exactly one word pair can end the sentence, etc. We leave out the details of these constraints here and refer the interested readers to the original paper.

Besides the sequential constraints, there are two more kinds of constraints: length constraints and syntactic constraints. Length constraints force the lengths of the generated sentences to fall into a value range. They are shown as follows:

$$\sum_{i=1}^n \delta_i \geq \text{minLength}, \quad \sum_{i=1}^n \delta_i \leq \text{maxLength},$$

where *minLength* and *maxLength* are minimum and maximum lengths for the compression and δ_i denotes the binary label of w_i in the sentence.

Syntactic constraints are a hard way to force the compressed sentence to satisfy some manually defined rules. For example, if a non-clausal modifier (**ncmod**) (such as an adjective or a noun) or determiners (**detmod**) is included in the compression, then the head of the modifier must also be included and at least one verb should be included in the compression. Generally, these constraints are based on syntactic analyses that are not guaranteed to be correct.

The ILP method searches for the best compression in the space of all possible compressions. Ideally, the result would be optimal if the computational time is not a concern. However, for long sentences, efficiency could be important. Furthermore, the length constraints and syntactic constraints are hard constraints, making it impossible to explore other possible compressions. The situation could become worse when there are parsing errors. In the following section, we propose our reinforcement learning framework, which runs faster and uses the length and syntactic constraints in a soft manner.

3.3 Our Reinforcement Learning Based Method

Reinforcement Learning (RL) is a powerful method of solving planning problems, especially problems which can be formulated as a Markov decision process (MDP) [17]. In RL, an agent tries to achieve a final target after a series of actions, where the actions are executed based on observations of the environment. At each step, there is an immediate reward encouraging or punishing the action. Four elements (state, action, reward and transition) make up the basic elements of an RL framework. Given an agent, the following steps describe how to solve MDP problems by RL:

1. The state s from the environment is observed by the agent.
2. An action is executed according to the current policy. The agent receives a reward. After the action, the state changes based on the transition function and **the new state** is observed again by the agent.
3. The agent repeatedly execute actions, receive rewards and pass into next states until it reaches the terminal condition.

In this section, we propose our DQN-based method for sentence compression. We model the sentence compression task as an MDP, where the model learns to delete words step by step based on the current state. We represent the MDP as a tuple (S, A, R, T) . Here S denotes the space of all possible states, A is a set of available actions, R denotes the reward function and $T(s'|s, a)$ is the transition function.

States. Each state in S in our MDP is simply a list of words $W = (w_1, w_2, \dots, w_n)$, where w_n is a special word *END* added by us. For each sentence, the initial state includes the original words of the sentence with *END* in the end.

Actions. At each step, the agent should perform either a *delete_i* action or a *finish* action, where *delete_i* deletes the i -th word from the current state while *finish* ends the MDP.

Rewards. The agent receives a reward each time an action has been executed based on the current state. If the current state is s_t , after executing action a_t , the agent receives the reward r_t as defined below:

$$r_t = \begin{cases} Sc(s_t) + \lambda m R_p & (a_t = \textit{finish}, \textit{slen} \in [\textit{minLength}, \textit{maxLength}], \\ R_p + \lambda m R_p & (a_t = \textit{finish}, \textit{slen} \notin [\textit{minLength}, \textit{maxLength}], \\ 0 & (\textit{otherwise}), \end{cases} \quad (2)$$

where $Sc(s_t)$ is a score function mapping a state to a numerical value, λ is a binary value controlling syntactic rewards, \textit{slen} is the length of word list in state, m is the number of constraints broken by the current compression and R_p is a negative value for punishment.

From the reward function, the agent only receives a reward if the action is *finish*. In order to have a fair comparison with the ILP method presented earlier, we use a trigram language model based scoring function for $Sc(s_t)$:

$$S(s_t) = p(x_0|\textit{start}) + \sum_{i=2}^{n-2} p(x_i|x_{i-2}, x_{i-1}) + p(\textit{end}|x_{n-1}, x_n), \quad (3)$$

where x_i is the i -th word in the current state s_t . Please note that this objective function is the same with the previous ILP method shown in Eq. 1.

In the ILP method, syntactic rules are used as hard constraints, making some possible compressions outside of the search space. In our model, we use reward

in a soft manner to consider these constraints. Here λ in the reward function is used to control the syntactic rules. When λ is set to 0, the model does not consider syntactic rewards. On the other hand, if λ is set to 1, syntactic rewards are considered in the reward function.

Similar to ILP method, we consider the following syntactic rules: (1) If a non-clausal modifier or determiner is included, its head should also be included. (2) If a head is included in the compression, then the children with type *neg* or *possessive* should also be included. (3) At least one verb should be included and verbs should be included along with their subjects and objects. (4) For prepositional phrases and subordinate clauses, head words should be included if one of the modifiers is included. It is also true reversely, i.e., at least one of the modifiers should be included if the head word is included. (5) For coordination, if two head words are included in the compression, then the coordinating conjunction must also be included. When calculating the reward, we count the number of rules broken by a result and multiply it by the punishment value R_p . The result value is added to the score for punishment.

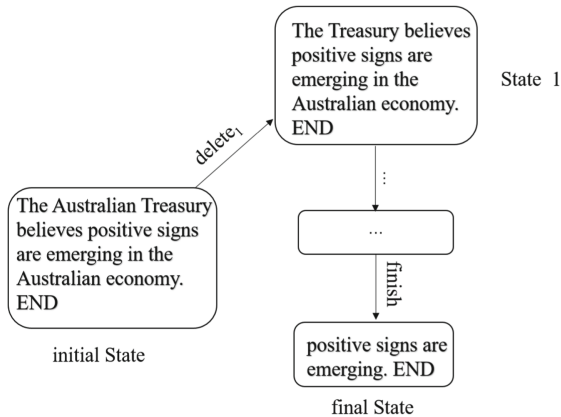


Fig. 1. The MDP for sentence compression with a specific example: the agent delete *Australian* at the first step (index starts from zero), by this analogy, with a sequence of actions, the sentence “The Australian Treasury believes positive signs are emerging in the Australian economy” is compressed to “positive signs are emerging”.

Transitions. In each episode, the MDP starts with a state s , i.e., a list of words. The MDP repeatedly deletes words in the list. The transition function $T(s'|s, a)$ incorporates the current state s with an action a and produces the new state s' . Whenever a is the *finish* action, the process stops. The MDP for sentence compression can be illustrated by Fig. 1 with a specific example.

DQN for Sentence Compression. The MDP described in the previous section can be seen as a sequence of transitions (s, a, r, s') . A state-action value

function $Q(s, a)$ is utilized by the agent to determine which action a should be performed for state s . This value function could be learned by Q-learning [22] in an iterative manner. The Bellman Equation [17] is utilized to calculate the target Q value each time:

$$Q_i(s, a) = E[r + \gamma \max_{a'} Q_{i+1}(s', a') | s, a]. \quad (4)$$

Here, r is the immediate reward after the agent takes action a on state s and γ is a discounting factor of future rewards.

In our study, we use a DQN [12] to model the value function. For a state $s = W$, we assume each word in W is represented by a d -dimensional vector. A standard bi-directional LSTM model [4] is utilized to encode the word embeddings sequentially in two directions. We can get a sequence of hidden vectors $(\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_n)$ and cell state vectors $(\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_n)$, where $\mathbf{h}_i \in \mathbb{R}^h$ and $\mathbf{c}_i \in \mathbb{R}^h$ as followings:

$$\begin{aligned} \vec{\mathbf{h}}_i, \vec{\mathbf{c}} &= \text{LSTM}_{\vec{\Theta}}(\vec{\mathbf{h}}_{i-1}, \mathbf{w}_i), \\ \overleftarrow{\mathbf{h}}_i, \overleftarrow{\mathbf{c}} &= \text{LSTM}_{\overleftarrow{\Theta}}(\overleftarrow{\mathbf{h}}_{i+1}, \mathbf{w}_i), \\ \mathbf{h}_i &= \vec{\mathbf{h}}_i \oplus \overleftarrow{\mathbf{h}}_i, \\ \mathbf{c}_i &= \vec{\mathbf{c}}_i \oplus \overleftarrow{\mathbf{c}}_i, \end{aligned}$$

where \oplus represents concatenation of vectors, and $\vec{\Theta}$ and $\overleftarrow{\Theta}$ are parameters of the bi-LSTM model (Fig. 2).

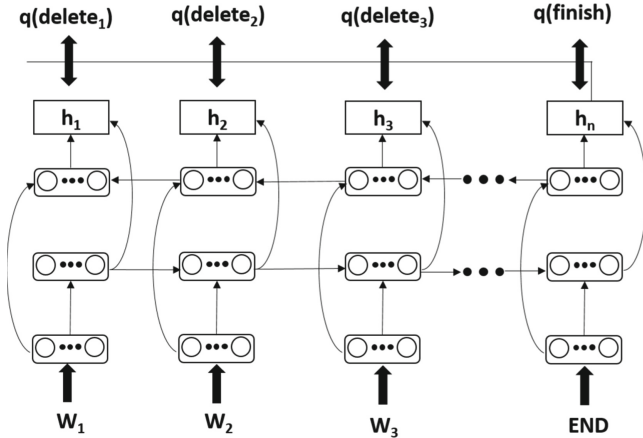


Fig. 2. Our DQN model for sentence compression

The concatenated vector \mathbf{c}_n then is used to calculate Q values as follows:

$$\mathbf{q}_j = \mathbf{V}^T \tanh(\mathbf{W}_1 \mathbf{x}'_j + \mathbf{W}_2 \mathbf{c}_n), j \in (1, \dots, n),$$

where q_j is the Q value for the action $delete_j$ or $finish$, and \mathbf{V} , \mathbf{W}_1 , and \mathbf{W}_2 are learnable parameters of the model. This is motivated by the pointer neural network [20]. If the maximum value points to the special token END , it means we should take the $finish$ action to terminate the process; otherwise we should take the $delete_j$ action and step into the next state.

The parameters of the DQN were trained by Adam [8]. Each time the parameters are updated to close the gap between the Q value predicted by DQN and the Q-value calculated by the Bellman Equation. We use an experience replay memory D to store transitions. The parameters are updated by minimizing the loss function shown below with a batch of transitions randomly sampled from D :

$$Loss(\theta) = (y - Q(s, a; \theta))^2, \quad (5)$$

where y is the target Q value calculated by the Bellman Equation. Algorithm 1 details the DQN training procedure.

Algorithm 1. The training Procedure for the DQN agent.

Initialize parameters θ and experience memory D

for $epoch \in [1, n]$ **do**

for $sentence \in train$ **do**

 initial state $s \leftarrow (x_1, x_2, \dots, x_n)$

for iteration $i=1, sentence\ length$ **do**

if $random() < \varepsilon$ **then**

 select a random action a_i

else

 compute $Q(s_i, a)$ for all actions using DQN; select $a_i = \arg \max Q(s_i, a)$

end if

 execute the action a_i , get the reward r_i , run into next state s_{i+1} ;

 store transition tuple $[s_i, a_i, r_i, s_{i+1}]$ in D ;

 sample random mini batch of transitions (s_j, a_j, r_j, s_{j+1}) from D ;

$$y_j = \begin{cases} r_j, & \text{if } s_{j+1} \text{ is } END \\ r_j + \gamma \max_{a'} Q(s_{j+1}, a'; \theta_t), & \text{otherwise} \end{cases} \quad (5)$$

 Perform gradient descent on the loss: $(y_j - Q(s_j, a_j; \theta))^2$;

if a_i is $finish$ **then**

 break

end if

end for

end for

end for

4 Experiment

In this section, we first introduce the datasets used to evaluate our methods and baselines. Then we present the results and some discussions.

4.1 Datasets and Experiment Setting

As we are more interested in unsupervised methods, we do not need any labeled data for training. In order to test the performance, we use the following datasets.

GoogleNews: The data¹ contains 10,000 sentence pairs collected and released by [2]. It was automatically obtained from the web through Google News.

BNCNews: The data that includes 1,500 sentence pairs was collected from the British National Corpus (BNC) and the American News by [1]. Annotators were asked to manually annotate the sentences. These annotations were only used as ground truth to evaluate the performance of different methods.

BroadCast: This dataset² was also collected by [1] from the English Broadcast News corpus. It contains about 1,400 sentence pairs. It is about spoken English and manually labeled. For each sample, three different annotators were asked to label it. We calculate the average score on the three annotations when evaluating the results.

We evaluate our methods in two settings as follows:

PRL: This is our pointer RL framework described previously without syntactic rewards. To do this, λ is set to 0.

PRL + Syn: In this setting, we add soft syntactic rewards on top of **PRL** by setting λ to 1.

In order to train the policy, we need some unlabeled data. We crawled 10,000 sentences for training and 1,000 sentences for validation on the Web from Google news. When training the DQN, we select the parameters with the best reward score on the validation data. In our experiments, word embeddings are initialized with GloVe 50-dimensional pre-trained embeddings [15]. The dimension of the hidden layers of bi-LSTM is 100. The size of experience memory D is set as 10,000 and each time sample batch size is 30. Discount factor for future reward is 0.9 and the punishment reward R_p is set as -0.1 . For minLength and maxLength in our method and ILP method, we set the values to twenty percent and eighty percent length of the corresponding source sentence respectively.

We compare our methods with following baselines:

ILP: This is the ILP method proposed by [1]. We evaluated two versions introduced in previous part: one without using syntactic constraints and the other with syntactic constraints. They are denoted as **ILP** and **ILP + Syn**, respectively.

Seq2seq: This is a sequence-to-sequence method proposed by [2]. As we do not have labeled data for training, we use the **ILP** method to generate the auxiliary labels on the data we collected for training DQN first, then trained the model by the auxiliary labels. We also use **ILP + Syn** to generate auxiliary labels and thus the model trained on the auxiliary labels is denoted as **Seq2seq + Syn**.

¹ available at <http://storage.googleapis.com/sentencecomp/compression-data.json>.

² BNCNews and BroadCast are available at <http://jamesclarke.net/research/resources/>.

Greedy: This method is a simple greedy algorithm, which repeatedly deletes words by maximizing the same reward function shown in Eq. 2. Each time the method compares the rewards of all the actions and selects the one with the highest score. Also, we can consider the syntactic rewards in the same way as **PRL + Syn**. We use **Greedy + Syn** to refer to it.

In the experiments, we use a pre-trained trigram language model³ to score the sentences for both our methods and the baselines. All the code is implemented in python and all the experiments are run on the same computer with an Intel(R) Core(TM) i7-5820K CPU @ 3.30 GHz.

4.2 Evaluation and Discussion

We evaluate two parts of the results: computational time and sentence compression performance. For the computational time, we measure the time cost (tc) on each dataset for different methods. When evaluating methods with syntactic features, we also add the time cost in syntactic analysis. For the compression performance, following previous work [2, 21], we report word level accuracy (acc) and F1 score for the retained words (flr). Besides, we also add F1 score of ROUGE-2 evaluation (r2f) [10]. The results on the three datasets can be seen from Table 1.

Table 1. Evaluation of our sentence compression methods.

	GoogleNews				NBCNews				BroadCast			
	ct(s)	acc(%)	flr(%)	r2f(%)	ct	acc	flr	r2f	ct	acc	flr	r2f
ILP	6823.90	51.04	49.39	25.64	2235.06	53.70	62.94	33.82	918.03	53.06	62.73	32.18
ILP + Syn	8547.56	51.33	51.57	29.84	2554.30	55.54	65.42	38.63	1021.50	55.07	65.08	37.29
Seq2Seq	248.02	49.32	47.44	20.07	42.02	50.29	58.79	28.54	35.39	51.46	60.01	25.36
Seq2Seq + Syn	528.63	50.85	49.82	25.99	87.26	52.84	62.12	35.30	64.88	54.68	62.23	33.20
Greedy	78.90	45.95	40.87	23.34	21.19	51.67	59.23	41.08	8.4	54.32	62.92	44.78
Greedy + Syn	298.89	46.59	41.48	22.50	55.72	52.19	59.55	40.79	28.86	55.34	63.81	45.20
PRL	520.17	52.49	55.08	33.40	85.15	56.21	67.49	39.59	59.47	54.64	66.27	36.17
PRL + Syn	737.87	53.43	56.52	34.41	105.15	56.69	66.41	39.64	79.47	55.49	66.54	38.85

From Table 1, In terms of time cost, we found greedy search was the fastest. Seq2seq and our method are much faster than the ILP method. For the compression performance, through accuracy (acc), F1 on retained words (flr) and ROUGE-2 F1 (r2f), most of the time, our method achieved the best results.

A question one may ask is why our method could perform better than the ILP method, which is globally optimized and thus should give the best results. Besides, some other results also appear to be strange. For example, the greedy method achieved the highest F1 score of ROUGE-2 on both NBCNews and BroadCast. We believe that this is because the objective function we use in the optimization problem is to approximate what we would like to achieve and does not necessarily always correlate with the actual performance. In order to verify

³ Accessible from www.keithv.com/software/csr/.

this, we calculated the average value of the objective function for the results of ILP, seq2seq, greedy and PRL. It is shown in Table 2. From the average value of the objection function, we can see that undoubtedly ILP method still achieved the highest value.

Table 2. Average object function score across datasets

	GoogleNews	NBCNews	BroadCast
ILP	1.64	1.74	1.07
Seq2Seq	0.72	0.79	0.53
Greedy	0.65	0.68	0.47
PRL	1.22	1.33	0.75

We also evaluate the time cost along with sentence length using sentences in the three datasets. Figure 3 is the scattergram for different methods without syntactic features⁴. Among these methods, the ILP method is the slowest which has an exponential time complexity with respect to sentence length. This is easy to understand because the ILP method theoretically should search in a space with a size of 2^n where n is the number of words in a sentence. For our method, at each time an action is selected from n -candidates and the action is taken at most n times, so the worst case time complexity should be n^2 . The seq2seq method has a constant time complexity which means sentence length may not affect the computational time. On the contrary, the greedy search method is sensitive to sentence length.

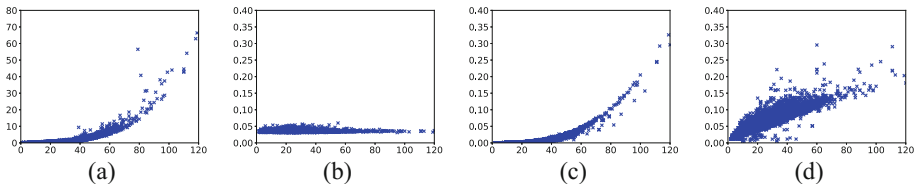


Fig. 3. Time consuming for different sentence length of different methods: (a) ILP (b) Seq2Seq (c) Greedy (d) PRL. The x -coordinate is the sentence length in words and the y -coordinate is the running time in terms of seconds. Note that the value range of y -coordinates in the (a) is $(0,80)$ which is different from others with $(0,0.4)$.

5 Conclusion

In this work, we explore how to use a deep reinforcement learning framework for sentence compression. We formulate the task as a Markov decision process

⁴ Methods with syntactic features have a similar pattern.

by deleting words step by step. Each time the agent decides to either delete a word or stop the compression. Besides, we use the hard constraints in previous methods in a soft manner. The experiment results showed that our method could achieve a comparable performance with a global optimal ILP method with much less time consumed.

References

1. Clarke, J., Lapata, M.: Global inference for sentence compression: an integer linear programming approach. *J. Artif. Intell. Res.* **31**, 399–429 (2008)
2. Filippova, K., Alfonseca, E., Colmenares, C.A., Kaiser, L., Vinyals, O.: Sentence compression by deletion with LSTMs. In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* (2015)
3. Filippova, K., Strube, M.: Dependency tree based sentence compression. In: *Proceedings of the Fifth International Natural Language Generation Conference* (2008)
4. Graves, A., Jaitly, N., Mohamed, A.R.: Hybrid speech recognition with deep bidirectional LSTM. In: *2013 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pp. 273–278. IEEE (2013)
5. Grissom II, A.: Don’t until the final verb wait: reinforcement learning for simultaneous machine translation (2014)
6. Hori, C., Furui, S.: Speech summarization: an approach through word extraction and a method for evaluation. *IEICE Trans. Inf. Syst.* **87**(1), 15–25 (2004)
7. Jing, H.: Sentence reduction for automatic text summarization. In: *Proceedings of the Sixth Conference on Applied Natural Language Processing* (2000)
8. Kingma, D., Ba, J.: Adam: a method for stochastic optimization. In: *Proceedings of the International Conference on Learning Representations* (2015)
9. Knight, K., Marcu, D.: Statistics-based summarization-step one: sentence compression. In: *Proceedings of the 17th National Conference on Artificial Intelligence* (2000)
10. Lin, C.Y.: ROUGE: a package for automatic evaluation of summaries. In: *Text Summarization Branches Out: Proceedings of the ACL-04 workshop*, vol. 8. Barcelona, Spain (2004)
11. McDonald, R.T.: Discriminative sentence compression with soft syntactic evidence. In: *Proceedings of European Chapter of the Association for Computational Linguistics Valencia* (2006)
12. Mnih, V., et al.: Playing atari with deep reinforcement learning. *arXiv preprint [arXiv:1312.5602](https://arxiv.org/abs/1312.5602)* (2013)
13. Narasimhan, K., Yala, A., Barzilay, R.: Improving information extraction by acquiring external evidence with reinforcement learning. *arXiv preprint [arXiv:1603.07954](https://arxiv.org/abs/1603.07954)* (2016)
14. Paulus, R., Xiong, C., Socher, R.: A deep reinforced model for abstractive summarization. *arXiv preprint [arXiv:1705.04304](https://arxiv.org/abs/1705.04304)* (2017)
15. Pennington, J., Socher, R., Manning, C.D.: GloVe: global vectors for word representation. In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (2014)
16. Ryang, S., Abekawa, T.: Framework of automatic text summarization using reinforcement learning. In: *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pp. 256–265. Association for Computational Linguistics (2012)

17. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction, vol. 1. MIT press, Cambridge (1998)
18. Turner, J., Charniak, E.: Supervised and unsupervised learning for sentence compression. In: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, pp. 290–297. Association for Computational Linguistics (2005)
19. Vandeghinste, V., Pan, Y.: Sentence compression for automated subtitling: a hybrid approach. In: Proceedings of the ACL workshop on Text Summarization, pp. 89–95 (2004)
20. Vinyals, O., Fortunato, M., Jaitly, N.: Pointer networks. In: Advances in Neural Information Processing Systems, pp. 2692–2700 (2015)
21. Wang, L., Jiang, J., Chieu, H.L., Ong, C.H., Song, D., Liao, L.: Can syntax help? Improving an LSTM-based sentence compression model for new domains. In: Meeting of the Association for Computational Linguistics, pp. 1385–1393 (2017)
22. Watkins, C.J., Dayan, P.: Q-learning. *Mach. Learn.* **8**(3–4), 279–292 (1992)



A Biomedical Question Answering System Based on SNOMED-CT

Xinhua Zhu, Xuechen Yang, and Hongchao Chen^(✉)

Key Lab of Multi-Source Information Mining and Security,
College of Computer Science and Information Engineering,
Guangxi Normal University, Guilin 541004, China
chen7297@sina.com

Abstract. Biomedical question answering system is an important research topic in biomedical natural language processing. To make full use of the semantic knowledge in SNOMED-CT for clinical medical service, we developed a biomedical question answering system based on SNOMED-CT, which has the following characteristics: (a) this system takes the semantic network in SNOMED-CT as a knowledge base to answer the clinical questions posed by physicians in natural language form, (b) a multi-layer nested structure of question templates is designed to map a template into the different semantic relationships in SNOMED-CT, (c) a template description logic system is designed to define the question templates and tag template elements so as to accurately represent question semantics, and (d) a textual entailment algorithm with semantics is proposed to match the question templates in order to consider both the flexibility and accuracy of the system. The experimental results show that the overall performance of the system has reached a high level, which can give 85% of the correct answer and be used as a biomedical question answering system in a real environment.

Keywords: Question answering system · SNOMED-CT · Template matching

1 Introduction

As an important research field in natural language processing, Question-Answering (QA) systems are advanced information retrieval systems that can answer questions formulated by users in natural language form. Based on the domains to which they are oriented, QA systems can be classified into two types [1]: open domain-oriented systems and restricted domain-oriented QA systems. For open domain-oriented QA systems (e.g., Google), no domain restriction is placed on topics, and these systems can accept questions of topics in any domain. While restricted domain-oriented QA systems can only accept questions about topics in a certain domain such as biomedicine [1]. Biomedicine is a knowledge-intensive discipline; physicians often have many difficult problems in clinical settings and need extra help. For example, Ely and colleagues observed that physicians did not pursue answers to 45% of the 1062 questions posed in clinical settings [2]. Therefore, it is very important to study the biomedical question-answering system to answer the clinical questions posed by physicians [3–6].

Currently, according to the source of information, the biomedical QA system can be classified into two types: document-based QA system and ontology-based QA system. Early document-based QA system mainly searches answer information based on keyword technology, such as MedQA [3]. Currently, it has incorporated various semantic technologies to improve retrieval accuracy, such as AskHERMES [4], MiPACQ [5] and MEANS [6]. Ontology-based QA system extracts the answer directly from an ontology [7] or database [8], such as enquireMe [9] and AskCuebee [10]. The forms of questions in the document-based QA systems are more flexible and diverse; while the answers provided by the ontology-based QA systems are more accurate and professional.

The concept of an ontology, which originated in philosophy, has been widely used in information science in recent years. An ontology is a unanimous agreement on shared concepts [11]. Using knowledge sharing as its core concept, ontologies have been widely used in knowledge engineering from the beginning [12]. Due to the importance of knowledge representation and terminology, the biomedical field has been very inclined to define a structured thesaurus or ontology to achieve global domain knowledge sharing [13], such as the Unified Medical Language System (UMLS) [14] of the National Library of Medicine (NLM). One of the largest and most extensive sources included in UMLS is SNOMED-CT [15] (Systematized Nomenclature of Medicine Clinical Terms). It is a structured comprehensive clinical terminology based on concepts and relations between concepts, which is developed and maintained by the International Health Terminology Standards Development Organization (IHTSDO). Nowadays, it contains more than 440,000 active concepts and over 1 million active relational records. More importantly, it is still in continuous renewal and evolution (updated four times a year). According to their clinical manifestations, the activity of concepts and relationships in SNOMED-CT is adjustable. Currently, based on its “is-a” taxonomy, SNOMED-CT is widely used in the semantic similarity measurement of biomedical terms [16] and data capture and decision support [17, 18]. However, in addition to the is-a relationship, SNOMED-CT also contains more than 70 other semantic relationships such as *finding site* and *associated morphology*, and a large number of records of these relationships. Therefore, SNOMED-CT is not only a systematic taxonomy of medical terms, but also a medical clinical knowledge base based on a semantic network. Unfortunately, its clinical knowledge based on the semantic network has not been fully exploited for use in clinical medicine.

To make full use of the semantic knowledge in SNOMED-CT for clinical medical services, we developed a biomedical question answering system based on SNOMED-CT. This biomedical QA system takes the semantic network in SNOMED-CT as a knowledge base to answer the clinical questions posed by physicians in natural language form and employs the multi-layer nested question templates with description logics to map a template into the different semantic relationships in SNOMED-CT. Moreover, the system also adopts a textual entailment algorithm with semantics to match the question template in order to consider both the flexibility and accuracy of the system.

2 Overview of Question Answering System

The question answering system in this paper is an intelligent assistant consultation system related to medical common-sense. The SNOMED-CT medical ontology is mainly used as a knowledge base to answer user's common-sense questions in clinical medicine. The system mainly consists of four parts: SNOMED-CT medical ontology, the question template library, the question pre-processing module and the answer retrieval module. Their relationship is shown in Fig. 1.

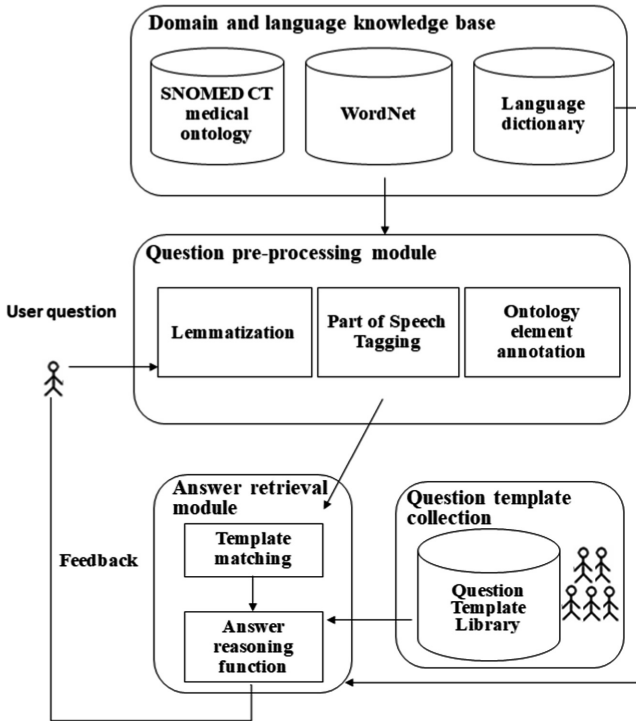


Fig. 1. Biomedical common-sense question answering system architecture diagram

SNOMED-CT Medical Ontology. The knowledge base of the question answering system in this paper is mainly composed of three parts: concept, description and relationship [19]. Among them, each concept has a uniquely readable Fully Specified Name (FSN), which indicates what the concept represents. In addition, from the description of this section to illustrate the concept, it is likely that a set of descriptions will appear with different terms to express the same concept. These descriptions support alternative expressions like multiple synonyms, and each synonym can express the same meaning. From the perspective of relations to express concepts, a set of relationships between a concept and other concepts express the logical definition of the concept that can be processed by the computer. In terms of description, this part of the

term is composed of multiple vocabularies, multiple phrases, and other readable strings that express the concept. Each description associates readability terms with a concept. For example, *Pimple (morphologic abnormality)* is a fully specified name of a concept that represents the pimple seen from the pathologist at the organizational level; The concept whose concept ID is 38521007 has synonyms *Displacement of tooth*, *Transposition of tooth*, *Reverse position of adjacent teeth* and *Tooth transposition*. In addition, each description has a unique description identifier and is arranged as a row in the description table. Relationships represent the connection between two concepts. This connection specifies three concepts, the source concept, the destination concept, and the relation type. For example, a relation typeid is 363698007, whose terminology description is *finding site*; source concept ID is 10000006, whose terminology description is *Radiating chest pain (finding)*; destination concept ID is 51185008, whose terminology description is *Thoracic structure (body structure)*. It expresses the relationship meaning that the finding site of *Radiating chest pain* is *Thoracic structure*.

WordNet. This paper applies for the ontology knowledge base that improves the matching accuracy in the process of template matching. WordNet [20] is a product of a research project at Princeton University and is a large online English classification database system that organizes lexical information according to its meaning. WordNet is gradually becoming an international common standard, and many countries are planning to establish a local language WordNet system compatible with English WordNet. Through a variety of semantic relationships, nouns, verbs, adjectives, and adverbs in WordNet are organized into a semantic web of synonym sets. The relationship between words in WordNet includes synonymy relationship, antisense relationship, hypernym relationship, hyponym relationship, overall relationship, partial relationship, approximate relationship, implication relationship and causality relationship. Each word in WordNet contains three important elements: concept, ID code, and gloss. A concept is a set of synonyms, and a word can correspond to multiple concepts. An ID code is a tag used to uniquely identify a concept. Concepts and ID codes are in a one-to-one relationship. ID codes between concepts are not repeated. A gloss is a one-line comment on a concept. A concept can correspond to multiple glosses.

Question Template. The question template in the question answering system of this paper is a question formula based on ontology and logic. It can realize the maximization of the semantics of the question and avoid the partial semantic loss in the question triplet representation [21, 22]. At the same time, by combining the first-order logic to represent ontology elements, the semantic accuracy of the symbol-based ontology element representation [23, 24] can be improved. It consists of five structures of bindings: template structure, synonymous structure, nested structure, inference rules and inference functions. The collection of various representative question templates forms the question template library in the question answering system of this paper. It reflects the interest of the users and medical personnel in the related medical common-sense.

Question Pre-processing. This process includes word tagging, lemmatization and ontology annotation for user’s questions. Part-of-speech tagging refers to the process of identifying the part of speech for each word in a question. Lemmatization refers to the process of restoring a vocabulary of any form to a general form that expresses complete semantics. This paper uses the Stanford POS Tagger to accomplish the part-of-speech tagging for the user’s question and uses Stanford CoreNLP [25] to accomplish lemmatization for the question. Then the ontology element annotation is the process of sequentially identifying the domain ontology concepts, relationships, and concept sets that appear in the user’s question. In this process, the conditional random field learning method [26] is used to recognize medical name entity in the user’s question. The first step is concept annotation. The medical named entities and nouns identified in the user’s question are matched with the concept and its synonyms in the SNOMED-CT medical ontology. If there is an identical concept, the noun is annotated as an ontology concept. The labelling format is: *<concept name: Concept>* , for example: *<Osteomalacia: Concept>* . Then, the step is relationship annotation. The verbs identify in the user’s question and the remaining nouns are matched with the related synonyms in the ontology. If there is an identical relationship, the verb or noun is marked as a relationship, and the labelling format is: *<Relationship name: Relation>* , for example: *<has focus: Relation>* . Finally, the step is concept set annotation. A plurality of concept names connected by conjunctives or punctuation are combined into a concept set in a user’s question sentence, and the annotation format is: *<{concept set}: ConceptSet>* , for example: *<{Meningoencephalitis, Meningomyelitis, Pachymeningitis}: ConceptSet>* .

Template Matching. It is a process of matching the question input by the user with the question template in the question template library. In this paper, we use the text implication algorithm which combines the ontology element annotation with the word similarity based on WordNet to achieve the matching inference between the user’s question and the template. If the user’s question has a high degree of implication for a certain template in the question template library and exceeds the threshold, the user’s question is considered to match the question template. The template matching proposed in this paper can improve the robustness of the similarity matching algorithm based on text [23], and at the same time can improve the accuracy of the matching algorithm based on word [24].

Semantic Acquisition. Replacing the ontology element variables in the matched template with the ontological element constants in the user’s question sentence, so as to obtain the question semantics and answer semantics of the user’s questions through the matched question template.

Answer Extraction. The transferred ontology element by Semantic acquisition as parameters, reasoning the answer by calling the inference rule bound by the matched template, and then extract the relevant answer of the user’s question from the ontology.

3 Structure of Question Template

The question template in this paper is a question formula based on the SNOMED-CT medical ontology. It binds the template structure, synonymous structure, nested structure, inference rules and inference function to form a mapping from the user's question to the answer to the question, whose BNF is defined as:

$$\langle \text{question template} \rangle ::= (\langle \text{template structure} \rangle, \{ \langle \text{synonymous structure} \rangle \}, \langle \text{nested structure} \rangle, \langle \text{reasoning rule} \rangle, \langle \text{reasoning function} \rangle).$$

- (1) Question structure: refers to the syntactic structure of an input template and the shallow semantics of user's input that are represented with variables and tags.
- (2) Synonymous structure: refers to a question structure with the same semantics of the main structure. One input template may contain multiple synonymous structures.
- (3) Nested structure: refers to a mechanism that implements the mapping of a template to different semantic relationships. A template can contain multiple nested structures. Each nested structure consists of a synonymous question structure, a preferred inference rule and its inference function, multiple ordered inferential inference rules and their inference functions. Each inference rule is associated with a semantic function, where the preferred inference rule is closest to the semantics of the question structure of the nested structure. The system will determine which nested structure the user question belongs to in the template based on the match between the user's question and each nested structure of the template.
- (4) Reasoning rule: represents the deep semantics of an input template and the reasoning process of the expected answer or the intention of the user's input. The rule semantics is accurately represented by a domain ontology-based predicate formula.
- (5) Reasoning function: represents a template-bound program that performs the reasoning function specified by the reasoning rule.

Examples of template definition:

Question Template: Query disease symptoms

Related relations: *Associated morphology* (id = 116676008), *Associated with* (id = 47429007), *Due to* (id = 42752001), *Interprets* (id = 363714003), *Has interpretation* (id = 363713009)

<Template structure>::=<what> are the < symptoms | indications | indicants > for patients with <C:Concept> [?]
 <Synonymous structure>::=<what> are the <symptoms |indications | indicants> for <C:Concept> [?]
 <Synonymous structure>::=<What><causes | induces stimulates | makes | gets | haves>patients with <C:Concept>[?]
 <Nested structure>::=<what> are the <symptoms | indications | indicants> for <C:Concept> [?]
 <Preferred reasoning rule>::=
 $\forall c(\text{Concept}(c) \wedge \exists am,s(\text{Relation}(am) \wedge am.id = "116676008" \wedge \text{ConceptSet}(s) \wedge \text{Domain_Range_of_Rel}(c, s, am) \rightarrow \text{Answer}(s)))$
 <Preferred reasoning function>::= Reasoning_function (c ,am)
 <Secondary reasoning rule>::=
 $\forall c(\text{Concept}(c) \wedge \exists aw,s(\text{Relation}(aw) \wedge aw.id = "47429007" \wedge \text{ConceptSet}(s) \wedge \text{Domain_Range_of_Rel}(c, s, aw) \rightarrow \text{Answer}(s)))$
 <Secondary reasoning function>::= Reasoning_function1 (c ,aw)

There are multiple synonymous questions and nested structures with the same semantics as the question structure in question templates that ask questions about the symptoms of the disease. The order of the reasoning rules of each nested structure is determined by the situation of question matching. This can not only make the reasoning rules form correspondence with the higher matching questions, but also make the reasoning rules more diversified to obtain more answer information.

4 Template Matching

Template matching refers to selecting a matching question template for a pre-processed user's question from a template library, which is an important part of answer retrieval. In order to increase the robustness of template matching, this paper does not use a matching algorithm based on text similarity [23], and the matching computing between user's question and template is regarded as a textual solution process [27], which check whether the user's question contains the semantics of the template. In addition, when the question is matched, the dependence relationship of the question is analyzed first, and the affirmative and negative meanings of the user's question are analyzed, and then the template matching is performed according to the affirmation and the negation of the question. Based on the word-based matching inference algorithm, this paper further improves the match accuracy of word matching inference by expanding the annotation of ontologies and the similarity calculation based on WordNet.

This paper comprehensively considers the following two kinds of commonly used text implication algorithms [28], and draws conclusions based on word-based template matching inference.

(1) Simple Matching Algorithm

This method does not consider the continuity of words. Only if the user's question sentence completely includes the words in the question template, it is considered to match the template. The formula is:

$$spmatch(User, Temp) = \frac{\sum_{i \in T} Lmatch(i)}{|T|} \quad (1)$$

where User represents any user question, Temp represents any question template, and T is a set of mandatory words that the question template Temp contains, and is calculated as follows:

$$Lmatch(i) = \begin{cases} 1 & \text{if } \exists j \in U \ i = j, \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

where U represents the set of words contained in the user's question User.

(2) Consecutive subsequence matching

This method gives high attention to consecutive subsequences. This method will determine if the user's question contains any possible consecutive subsequences in the template, and the length of the consecutive subsequence ranges from two words to the sentence length of the entire question template. The specific formula is as follows:

$$LCSmatch(User, Temp) = \frac{\sum_{i=2}^{|T|} f(ST_i)}{|T| - 1} \quad (3)$$

where ST_i represents a consecutive subsequence of length i in the question template Temp, $f(ST_i)$ the formula is as follows:

$$f(ST_i) = \frac{\sum_{j \in ST_i} Lmatch(j)}{|T| - i + 1} \quad (4)$$

$$Lmatch(j) = \begin{cases} 1 & \text{if } \exists k \in SU_i \ k = j, \\ 0 & \text{otherwise.} \end{cases} \quad (5)$$

where SU_i represents that a consecutive subsequence of length i in user's question User.

Because the question template contains a large number of ontology element variables and their type specifiers, in order to improve the accuracy of matching, the user's question needs the annotation of the ontology element before the matching, and for the ontology element, the matching computing only if their types are the same, they are judged to be the same sentence elements.

In this paper, when we use the matching inference algorithm to perform template matching on user’s question sentences, we judge whether two words are the same, which are not only based on the same words, but also based on whether the two words are synonymous or very similar. We calculate word similarity based on WordNet to determine whether two words are synonymous. If the similarity between two words is higher than 0.75, we consider these two words to be synonymous. Then, the two words match. The similarity method is based on WordNet [17].

Moreover, we perform semantic analysis for user’s question dependence relationship and use Stanford Dependencies at Stanford University to perform a dependency syntax analysis for user’s questions to analyze the affirmative or negative meanings of user’s questions. If the user’s question asks for a positive meaning, then we let the question match the positive question in the question template library; otherwise match the negative question. This can greatly reduce the problem that the user’s question is not consistent with the user’s question expectation due to semantic problems, and more accurately meets the user’s answer expectations.

5 Experiments

We evaluate our question answering system from two aspects about template matching and system performance. The template matching evaluation is mainly the question classification performance in the answer retrieval module of the examination system, including the classification accuracy of various matching algorithms and the completeness of the template library. The assessment index has the precision of the question classification, the recall and the F1 value. We use F1 value as the final comprehensive assessment indicator. The system performance evaluation is to examine the overall performance of the system, including the performance of the answer retrieval module, and the completeness and accuracy of the domain knowledge in the domain ontology. The correct answer rate of the system is used as the final assessment index.

To evaluate the template matching submodule of the answer search module, we asked 10 users to issue a related question based on each of the 50 templates in the question template library. A total of 500 related questions were generated. We use these 500 user’s questions for the evaluation experiment of template matching inference. The questions given by the 10 users were based on relevant knowledge in the medical field, and there were no questions about the concepts and relationships in other fields. We use the following two matching inference methods for template matching evaluation experiments:

- (1) **Ontology-based Lexical Reference (LB) inference (LB + ontology):** The ontology elements are considered in the vocabulary match inference formulas (1) and (3) described in Sect. 4. Before user’s question was matched, the annotation of ontology element and lemmatization are performed. For ontology elements, they are determined to be the same vocabulary as long as their types are the same in the matching computing; the matching computing between the question and the template is as follows;

$$match(User, Temp) = \frac{spmatch(User, Temp) + LCSmatch(User, Temp)}{2} \quad (6)$$

if $match(User, Temp) \geq MTH$ push($CandStack, Temp$)

where the conditional statement indicates that if the degree of matching for a user's question with a certain template is greater than the specified matching threshold MTH , the template is pushed into the candidate template stack $CandStack$.

- (2) LB + ontology + inference based on question dependency semantics (LB + ontology + semantics): On the basis of Method 1, we further consider the matching inference based on the question dependency semantics introduced in Sect. 4.

If the degree of match between the multiple templates in the template library and the user's question is greater than the specified matching threshold, the template with the maximum matching degree is selected as the final matching template of the user's question. The question classification effect of the above two matching inference methods is shown in Fig. 2.

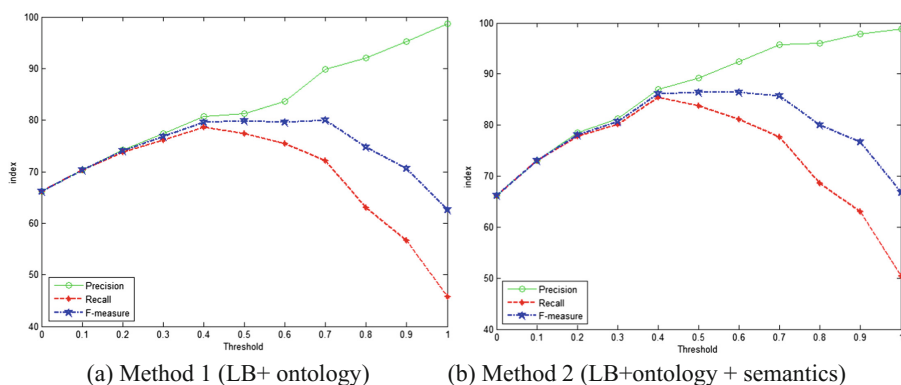


Fig. 2. Variation of the precision, recall rate, F1 value of the question classification with the matching threshold changes

Through the trend analysis of the above two graphs, we can obtain the best F1 value of each method and its corresponding matching threshold, precision and recall rate, as shown in Table 1:

Table 1. The best question classification index of the two methods

Method	Matching threshold	Precision	Recall	The best F1 value
LB + ontology	0.7	89.8	72.2	80.04
LB + ontology + semantics	0.6	92.48	81.2	86.47

To evaluate the overall performance of the restricted domain medical question answering system proposed in this paper, we asked 10 users who did not understand the system template library to ask questions about the system. Each person asked 10 questions about medical common-sense knowledge and asked the system to Answer it. We use the best matching threshold chosen for each method in the previous section. Then we use the following formula to evaluate the system performance accuracy and accuracy formula to obtain experimental results as shown in Table 2.

Table 2. System performance evaluation results

Method	Matching threshold	Accuracy	Correct rate	Correct answer	Wrong answer	Uncertain
LB + ontology	0.7	85.23	75	75	13	12
LB + ontology + semantics	0.6	90.43	85	85	9	6

6 Conclusions and Future Work

The experimental results in Table 1 show that the template matching inference method combined with semantic similarity can further improve the matching accuracy, so that the best classification F1 value can be obtained at a lower matching threshold (0.6), which can significantly improve the recall rate of question classification, laying the foundation for further improving the overall performance of the system. Moreover, the experiment obtained a higher optimal classification F1 value (86.47). It also shows that the template in our template library is relatively complete, which can basically meet the user's demand for medical common-sense.

The evaluation results in Table 2 show that when the system adopts the template matching inference method combining ontology and semantic similarity, it can give a correct answer of 85%. It shows that the overall performance of the system has reached a high level and can meet the user needs of intelligent answering to medical common-sense questions.

However, there are still many deficiencies in this paper that need to be improved. The main problems are: the knowledge in SNOMED-CT medical ontology is not complete enough, especially for more detailed information of some diseases is not comprehensive enough. Moreover, synonymous question structures contained in the question template database are not sufficient enough. This makes the degree of matching between some synonymous user questions and the corresponding question template low in the matching process, which reduces the performance of the system to some extent. For these deficiencies, we will further improve our system in future studies.

Acknowledgements. This work has been supported by the National Natural Science Foundation of China under the contract numbers 61462010 and 61363036, and Guangxi Collaborative Innovation Center of Multi-source Information Integration and Intelligent Processing.

References

1. Sun, C., Guan, Y., Wang, X., Wang, Q., Liu, T.: InsunTourQA: a restricted-domain question answering system. *J. Comput. Inf. Syst.* **3**(4), 1581–1590 (2007)
2. Ely, J.W., Osheroff, J.A., Chambliss, M.L., Ebell, M.H., Rosenbaum, M.E.: Answering physicians' clinical questions: obstacles and potential solutions. *J. Am. Med. Inf. Assoc.* **12**(2), 217–224 (2005)
3. Lee, M., Cimino, J., Zhu, H.R., Sable, C., Shanker, V., Ely, J., et al.: Beyond information retrieval—medical question answering. *Amia. Annu. Symp. Proc.* **2006**, 469–473 (2006)
4. Cao, Y., Liu, F., Simpson, P., Antieau, L., Bennett, A., Cimino, J.J., et al.: Askhermes: an online question answering system for complex clinical questions. *J. Biomed. Inf.* **44**(2), 277–288 (2011)
5. Cairns, B.L., Nielsen, R.D., Masanz, J.J., Martin, J.H., Palmer, M.S., Ward, W.H., et al.: The MiPACQ clinical question answering system. *AMIA. Annu. Symp. Proc.* **2011**, 171–180 (2011)
6. Abacha, A.B., Zweigenbaum, P.: MEANS: a medical question-answering system combining NLP techniques and semantic web technologies. *Inf. Process. Manag.* **51**(5), 570–594 (2015)
7. Ray, S.K., Singh, S., Joshi, B.P., Beach, J.E.: A semantic approach for question classification using wordnet and Wikipedia. *Pattern Recogn. Lett.* **31**(13), 1935–1943 (2010)
8. Popescu, A.M., Etzioni, O., Kautz, H.: Towards a theory of natural language interfaces to databases. In: *Proceedings of the 8th International Conference on Intelligent User Interfaces*, pp. 149–157. ACM New York (2003)
9. Wong, W., Thangarajah, J., Padgham, L.: Contextual question answering for the health domain. *J. Am. Soc. Inf. Sci. Technol.* **63**(11), 2313–2327 (2012)
10. Asiaee, A.H., Minning, T., Doshi, P., Tarleton, R.L.: A framework for ontology-based question answering with application to parasite immunology. *J. Biomed. Semant.* **6**(1), 31–56 (2015)
11. Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D., Patel-Schneider, P.F.: *The Description Logic Handbook: Theory, Implementation and Applications*, 2nd edn. Cambridge University Press, New York (2010)
12. Strassner, J.: *Handbook of Network and System Administration: Knowledge Engineering Using Ontologies*. Elsevier, Amsterdam (2008)
13. Harispe, S., Sánchez, D., Ranwez, S., Janaqi, S., Montmain, J.: A framework for unifying ontology-based semantic similarity measures: a study in the biomedical domain. *J. Biomed. Inform.* **48**(2), 38–53 (2014)
14. Humphreys, B.L., Lindberg, D.A.: The UMLS project: making the conceptual connection between users and the information they need. *Bull. Med. Libr. Assoc.* **81**(2), 170 (1993)
15. Wei, D., Helen, G.H., Perl, Y., Halper, M., Ochs, C., Elhanan, G., et al.: Structural measures to track the evolution of SNOMED CT hierarchies. *J. Biomed. Inf.* **57**(C), 278–287 (2015)
16. Zhu, X., Li, F., Chen, H., Peng, Q.: An efficient path computing model for measuring semantic similarity using edge and density. *Knowl. Inf. Syst.* **55**(1), 79–111 (2018)
17. Kim, H.Y., Park, H.A.: Development and evaluation of data entry templates based on the entity-attribute-value model for clinical decision support of pressure ulcer wound management. *Int. J. Med. Inf.* **81**(7), 485–492 (2012)
18. Liu, J., Lane, K., Lo, E., Lam, M., Truong, T., Veillette, C.: Addressing SNOMED CT implementation challenges through multi-disciplinary collaboration. *Stud. Health Technol. Inf.* **160**(2), 981–985 (2010)
19. SNOMED CT Technical Implementation Guide January 2015 International Release (US English). <https://confluence.ihtsdotools.org/display/DOC>

20. Miller, G.A., Fellbaum, C.: Semantic networks of english. *Int. J. Cogn. Sci.* **41**(1), 197–229 (1991)
21. Lopez, V., Uren, V., Motta, E., Pasin, M.: AquaLog: an ontology-driven question answering system for organizational semantic intranets. *J. Web Semant.* **5**(2), 72–105 (2007)
22. Dzikovska, M., Steinhäuser, N., Farrow, E., Moore, J., Campbell, G.: BEETLE II: deep natural language understanding and automatic feedback generation for intelligent tutoring in basic electricity and electronics. *Int. J. AIED.* **24**(3), 284–332 (2014)
23. Zhu, X.H., Cao, Q.H., Su, F.F.: A chinese intelligent question answering system based on domain ontology and sentence templates. *Int. J. Digit. Content Technol. Appl.* **5**(11), 158–165 (2011)
24. Wang, D.: Answering contextual questions based on ontologies and question templates. *Front. Comput. Sci-Chi.* **5**(4), 405–418 (2011)
25. Stanford CoreNLP. <https://stanfordnlp.github.io/CoreNLP/simple.html>
26. Mccallum, A., Li, W.: Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In: *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pp. 188–191. ACL, Stroudsburg (2003)
27. Bos, J., Markert, K.: Recognising Textual entailment with robust logical inference. In: *Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pp. 628–635. ACL, Stroudsburg (2006)
28. Ferrández, Ó., Micol, D., Muñoz, R., Palomar, M.: DLSITE-1: lexical analysis for solving textual entailment recognition. In: Kedad, Z., Lammari, N., Métails, E., Meziane, F., Rezgui, Y. (eds.) *NLDB 2007. LNCS*, vol. 4592, pp. 284–294. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-73351-5_25



Authorship Attribution for Short Texts with Author-Document Topic Model

Haowen Zhang, Peng Nie, Yanlong Wen, and Xiaojie Yuan^(✉)

College of Computer and Control Engineering, Nankai University, Tianjin, China
{zhanghaowen, niepeng, wenyanlong, yuanxiaojie}@dbis.nankai.edu.cn

Abstract. The goal of authorship attribution is to assign the controversial texts to the known authors correctly. With the development of social media services, authorship attribution for short texts becomes very necessary. In the earlier works, topic models, such as the Latent Dirichlet Allocation (LDA), have been used to find latent semantic features of authors and achieve better performance on authorship attribution. However, most of them focus on authorship attribution for long texts. In this paper, we propose a novel model named Author-Document Topic Model (ADT) which builds the model for the corpus both at the author level and the document level to figure out the problem of authorship attribution for short texts. Also, we propose a new classification algorithm to calculate the similarity between texts for finding the authors of the anonymous texts. Experimental results on two public datasets validate the effectiveness of our proposed method.

Keywords: Authorship attribution · Topic model · Short text

1 Introduction

Authorship attribution has attracted much attention over the last decades because of its important role in criminal law, military intelligence, and humanities research [1]. The most majority of such researches try to determine the authors of controversial long texts, such as books, papers and so on. A lot of statistical learning methods have been used for authorship attribution and achieve good performance. In the recent years, more content is in the form of the short message with the growing popularity of Internet-based communication facilities, which creates great interest both in theory and computation on short texts. Compared to long texts, short texts have a few words, unclear structure and irregular usage of words. These characteristics make authorship attribution for short texts difficult, and the approaches that work well on long texts cannot give the same performance on short texts. As a result, finding the author of short texts (e.g., emails [2], blogs [3], twitters [4]) has attracted many researchers.

Support Vector Machine (SVM) is widely used in text classification. However, when it comes to authorship attribution, it cannot achieve the same performance because the goal of authorship attribution is not only to find texts which are

similar in content but also to consider the authors' writing style behind the content.

Finding the semantic feature of the author behind the content of the texts is very important for authorship attribution. Conventional topic models, such as PLSA [5] and LDA [6] assume that a document is a mixture of topics, where a set of correlated words is considered to be selected from the same topic. After enough number of iterations for training the model, words that appear in the same issue will be more likely assigned to the same topic. Based on this idea, LDA-H [7] first uses LDA to address the problem of authorship attribution and achieves better performance than SVM.

With the development of social media, the sparsity of content in short texts brings a new challenge to authorship attribution. To solve it, most of the early works [3, 4] prefer to aggregate short texts into a lengthy pseudo-document and build a feature set for each author. However, they ignore the effect of each text for authorship attribution in this way.

In this paper, we propose a novel topic model named Author-Document Topic Model (ADT) for authorship attribution on short texts. We combine two level topic models Author-Topic Model (AT) and traditional LDA as the final model, which treats every word of each document equally at both two levels. Our motivation is based on the observation that when an author writes a document, the word frequency of a document varies apparently with authors of different writing style, and documents talked about different things will also affect the usage of words. To find the most probable author of the anonymous document, we propose a new classification algorithm to calculate the similarity between the training documents and the given document. Then the author of the most similar document will be assigned to the target document.

The main contributions of this paper include:

- (1) We propose a novel generative model ADT for authorship attribution on short texts. We train ADT at both author level and document level. On the one hand, we use ADT to deal with the sparsity of content in short texts by aggregating short texts into lengthy pseudo-document. On the other hand, we use traditional LDA to make full use of training texts to find the authors of the anonymous texts.
- (2) We propose a new classification algorithm for authorship attribution, which combines the influence of both authors and documents to calculate the similarity between the training document and the anonymous document.
- (3) Experiment evaluations on two real-world datasets Pan'11 [8] and Blog [9] demonstrate the effectiveness of our proposed method. Compared to the current state of the art, ADT obtains a 6.63% improvement on Pan'11 and a 7.66% improvement on Blog.

The remainder of this paper is organized as follows. In the next section, we introduce the related studies in authorship attribution. In Sect. 3, we propose our ADT model and classification algorithm. The experimental evaluation is described in Sect. 4. In Sect. 5, we draw our conclusions.

2 Related Work

Authorship attribution is a traditional problem which can date back to the end of 19th century. It can be divided into two categories: similarity-based methods and machine-learning-based methods [1].

In similarity-based methods, SCAP [10] is the simplest method which calculates the Jaccard similarity between a given text and the profile texts of authors to find the most similar author. The feature sampling method (FS) [3] thinks that they do not know which features of the corpus are important for authorship attribution and which are not, so they randomly choose certain features from the feature set every time for calculating the similarity between the anonymous text and all authors' profiles. After repeating this process k times, the anonymous text is assigned to the author whose profile is most similar to the given text for a certain fixed number of the k times.

In machine-learning-based methods, SVM and topic model achieve better performance than others. SVM is often used for authorship attribution for its proven effectiveness in text classification and stability in handling many features [11,12]. Schwartz et al. [13] select features with k -signature, then they combine the feature set with flexible patterns for distinguishing authors, which is applied for SVM.

However, SVM tends to assign texts with similar features to the same author, which we think is not enough for authorship attribution. To obtain the latent semantic features of authors, the topic model LDA is applied to authorship attribution, and they use the Hellinger distance for calculating the similarity between document topic distribution to get the most similar author [7]. And two disjoint topic sets (DADT) [14] are trained separately, and they obtain better performance than ever before. Recent researches [4,15] focus on authorship attribution with the authors whose writing style is changed over time.

To the best of our knowledge, most of the previous works [3,4,7,13] only focus on the semantic features of authors, and they usually aggregate short texts into a lengthy pseudo-document and build feature sets for authors. In this paper, we are inspired by DADT [14] which builds the model for the corpus at both author level and document level and propose ADT model. Our ADT makes a great improvement on DADT, and we achieve better performance on authorship attribution for short texts.

3 Author-Document Topic Model

In this section, we detail the proposed model ADT. Firstly, we give the detail of ADT. Secondly, we explain the model inference. Thirdly, we describe the classification algorithm to find the most similar authors of the given texts. Finally, we make a comparison between ADT and DADT.

3.1 Topic Extraction via ADT

The graphical representation of ADT is shown in Fig. 1. It can be divided into two parts: Author-Topic Model on the left and Document-Topic Model on the right.

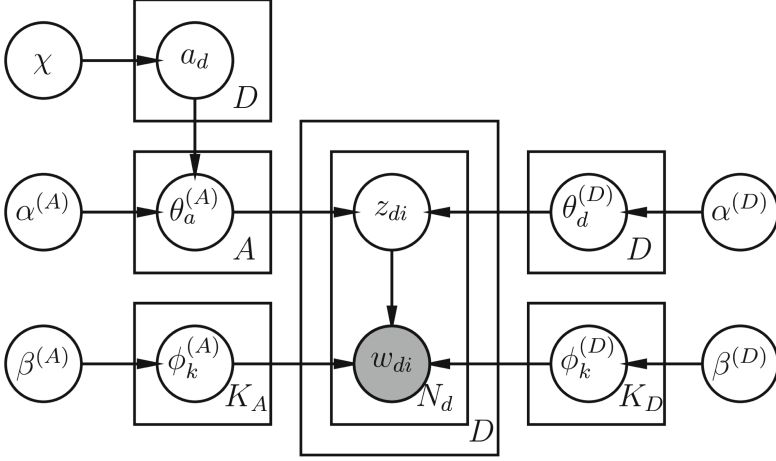


Fig. 1. The Author-Document Topic Model

We assume that the corpus has A authors, D documents, and V unique words in the vocabulary. \mathcal{D} and \mathcal{C} are used as the representation of the Dirichlet and categorical distributions respectively. Besides, K_A and K_D denote the number of topics of author level and document level. Formally, ADT assumes the following generative process for each document in a corpus \mathbf{D} :

Author Level

1. Draw an author distribution χ , which is determined by the number of documents for each author;
2. Draw an author topic distribution $\theta_a^{(A)} \sim \mathcal{D}(\alpha^{(A)})$ for each author a ;
3. Draw a word distribution $\phi_k^{(A)} \sim \mathcal{D}(\beta^{(A)})$ for each author topic k ;
4. Draw a word distribution $\phi_k^{(D)} \sim \mathcal{D}(\beta^{(D)})$ for each document topic k .

Document Level

1. Draw a document topic distribution $\theta_d^{(D)} \sim \mathcal{D}(\alpha^{(D)})$ for each document;
2. Draw document's author $a_d \sim \mathcal{C}(\chi)$.

Word Level

For each word i in document d :

1. Draw an author topic $z_{di}^{(A)} \sim \mathcal{C}(\theta_{a_d}^{(A)})$ and word $w_{di} \sim \mathcal{C}(\phi_{z_{di}^{(A)}}^{(A)})$;
2. Draw a document topic $z_{di}^{(D)} \sim \mathcal{C}(\theta_d^{(D)})$ and word $w_{di} \sim \mathcal{C}(\phi_{z_{di}^{(D)}}^{(D)})$.

3.2 Model Inference

Given the parameter $\theta_a^{(A)}$, $\theta_d^{(D)}$, $\phi_k^{(A)}$ and $\phi_k^{(D)}$, the conditional probability of word w_i is estimated as follows:

Author level:

$$\begin{aligned}
 p(w_i|\theta_a^{(A)}, \phi^{(A)}) &= \sum_{k=1}^{K_A} P(w_i, z_i = k|\theta_a^{(A)}, \phi^{(A)}) \\
 &= \sum_{k=1}^{K_A} P(z_i = k|\theta_{ak}^{(A)})P(w_i|z_i = k, \phi_{k,w_i}^{(A)}) \\
 &= \sum_{k=1}^{K_A} \theta_{ak}^{(A)} \phi_{k,w_i}^{(A)}.
 \end{aligned} \tag{1}$$

Document level:

$$\begin{aligned}
 p(w_i|\theta_d^{(D)}, \phi^{(D)}) &= \sum_{k=1}^{K_D} P(w_i, z_i = k|\theta_d^{(D)}, \phi^{(D)}) \\
 &= \sum_{k=1}^{K_D} P(z_i = k|\theta_{dk}^{(D)})P(w_i|z_i = k, \phi_{k,w_i}^{(D)}) \\
 &= \sum_{k=1}^{K_D} \theta_{dk} \phi_{k,w_i}.
 \end{aligned} \tag{2}$$

For all words, ADT maximizes the likelihood function for the corpus in two levels:

Author level:

$$p(\mathbf{D}|\theta_a^{(A)}, \phi^{(A)}) = \prod_{d=1}^D \prod_{n=1}^{Nd} \sum_{k=1}^{K_A} \theta_{ak}^{(A)} \phi_{k,w_i}^{(A)}. \tag{3}$$

Document level:

$$p(\mathbf{D}|\theta_d^{(D)}, \phi^{(D)}) = \prod_{d=1}^D \prod_{n=1}^{Nd} \sum_{k=1}^{K_D} \theta_{dk} \phi_{k,w_i}. \tag{4}$$

There are two common ways variational inference [6] and collapsed Gibbs sampling [16] for inferring topic models. We use collapsed Gibbs sampling to conduct approximate inference for $\theta_a^{(A)}$, $\theta_d^{(D)}$, $\phi_k^{(A)}$ and $\phi_k^{(D)}$. For i th word of d th document, its author is known as author a , ADT samples author topic $z_i^{(A)}$ according to the following conditional distribution:

$$p(z_i^{(A)} = k, x_{di} = a|z_{-i}^A, \mathbf{D}) \propto \frac{(n_{-i,k|a} + \alpha^{(A)})}{(n_{-i,*|a} + K_A \alpha^{(A)})} \frac{(n_{-i,w_i|k}^{(A)} + \beta^{(A)})}{(n_{-i,*|k}^{(A)} + V \beta^{(A)})}. \tag{5}$$

where $z_{-i}^{(A)}$ is the topic assignments for all words, $n_{-i,k|a}$ is the number of words assigned to author topic k in author a , $n_{-i,*|a}$ is the total number of words in author a , $n_{-i,w_i|k}^{(A)}$ is the number of word w_i in author topic k , and $n_{-i,*|k}^{(A)}$ is the total number of words in author topic k . All of them exclude current assignment of $z_i^{(A)}$.

In the same way, ADT sample document topic $z_i^{(D)}$ according to the following conditional distribution:

$$p(z_i^{(D)} = k | z_{-i}^{(D)}, \mathbf{D}) \propto \frac{(n_{-i,k|d} + \alpha^{(D)})}{(n_{-i,*|d} + K_D \alpha^{(D)})} \frac{(n_{-i,w_i|k}^{(D)} + \beta^{(D)})}{(n_{-i,*|k}^{(D)} + V \beta^{(D)})}. \quad (6)$$

where $z_{-i}^{(D)}$ is the topic assignments for all words, $n_{-i,k|d}$ is the number of words assigned to document topic k in document d , $n_{-i,*|d}$ is the total number of words in document d , $n_{-i,w_i|k}^{(D)}$ is the number of word w_i in document topic k , and $n_{-i,*|k}^{(D)}$ is the total number of words in document topic k . All of them same exclude current assignment of $z_i^{(D)}$.

After a sufficient number of iterations, we can estimate the topic attribution and word distribution as follows:

$$\theta_{ak}^{(A)} = \frac{n_{k|a} + \alpha^{(A)}}{n_{*|a} + K_A \alpha^{(A)}}, \quad (7)$$

$$\theta_{dk}^{(D)} = \frac{n_{k|d} + \alpha^{(D)}}{n_{*|d} + K_D \alpha^{(D)}}, \quad (8)$$

$$\phi_{kw}^{(A)} = \frac{n_{w|k}^{(A)} + \beta^{(A)}}{n_{*|k}^{(A)} + V \beta^{(A)}}, \quad (9)$$

$$\phi_{kw}^{(D)} = \frac{n_{w|k}^{(D)} + \beta^{(D)}}{n_{*|k}^{(D)} + V \beta^{(D)}}. \quad (10)$$

According to Eqs. (3), (4), the process that words are generated from our model is shown in Algorithm 1. Then we estimate author topic distribution and document topic distribution by Eqs. (7), (8) and estimate the topic word distribution for authors and documents by Eqs. (9), (10).

And the expected values for the corpus author distribution are:

$$\chi_a = \frac{1 + d_a}{A + D}. \quad (11)$$

where d_a is the number of documents belonging to the author a .

3.3 Authorship Attribution by Topic-Based Similarity

In the classification phase, we assume all test documents are written by a same unknown author, so no sampling would be required to obtain author topic distribution. Besides, we consider the word distribution of each document topic to be observed in the training phase and use Eq. (10) for getting its expected value. Then we carry on collapsed Gibbs sampling by the following conditional distribution for a given test document \tilde{d} :

$$p(z_i^{(D)} | z_{-i}, \tilde{\mathbf{D}}, \tilde{d}) \propto \frac{(n_{-i,k|\tilde{d}} + \alpha^{(D)})}{(n_{-i,*|\tilde{d}} + K_D \alpha^{(D)})} \phi_{k\tilde{w}_i}^{(D)}. \quad (12)$$

where $n_{-i,k|\tilde{d}}$ is the number of words assigned to document topic k in document \tilde{d} , and $n_{-i,*|\tilde{d}}$ is the number of words in \tilde{d} . All of them exclude current assignment of $z_i^{(D)}$. Finally, we will get test document's topic distribution by Eq. (8).

Algorithm 1. Topic assignment for words

Input:

- 1: K_A : the number of author topic;
- 2: K_D : the number of document topic;
- 3: $\alpha^{(A)}$: a single-valued hyper-parameter for $\theta^{(A)}$;
- 4: $\alpha^{(D)}$: a single-valued hyper-parameter for $\theta^{(D)}$;
- 5: $\beta^{(A)}$: a single-valued hyper-parameter for $\phi^{(A)}$;
- 6: $\beta^{(D)}$: a single-valued hyper-parameter for $\phi^{(D)}$;
- 7: \mathbf{D} : the training documents matrix;

Output: multinomial parameters $\theta^{(A)}$, $\theta^{(D)}$, $\phi^{(A)}$ and $\phi^{(D)}$;

- 8: **procedure** TOPIC ASSIGNMENT FOR WORDS
 - 9: Randomly initialize the topic assignments for all words;
 - 10: **while** not finished **do**
 - 11: **for** all documents $d \in [1, D]$ **do**
 - 12: **for** all words $w_{di} \in [1, N_d]$ in document d **do**
 - 13: Draw author topic $z_{di}^{(A)}$ by equation (5);
 - 14: Update $n_{k|a}$, $n_{w|k}^{(A)}$ and $n_{*|k}^{(A)}$;
 - 15: Draw document topic $z_{di}^{(D)}$ by equation (6);
 - 16: Update $n_{k|d}$, $n_{w|k}^{(D)}$ and $n_{*|k}^{(D)}$;
 - 17: **end for**
 - 18: **end for**
 - 19: **if** converged and L sampling iterations since last read out **then**
 - 20: read out parameter set $\theta^{(A)}$ according to Equation (7);
 - 21: read out parameter set $\theta^{(D)}$ according to Equation (8);
 - 22: read out parameter set $\phi^{(A)}$ according to Equation (9);
 - 23: read out parameter set $\phi^{(D)}$ according to Equation (10);
 - 24: **end if**
 - 25: **end while**
 - 26: **end procedure**
-

To find the author of test documents, we calculate the similarity between the test document \tilde{d} and the training document d based on the topic probability distribution as follows:

$$Similarity(\tilde{d}, d) = \chi_a \prod_{i=1}^{N_{\tilde{d}}} \sum_{k=1}^{K_A} \theta_{ak}^{(A)} \phi_{k\tilde{w}_i}^{(A)} - \sqrt{\sum_{k=1}^{K_D} (\tilde{\theta}_{\tilde{d}k}^{(D)} - \theta_{dk}^{(D)})^2}. \quad (13)$$

In the Eq. (13), we first calculate the probability of \tilde{d} written by the author of d . Then we calculate the Euclidean distance between \tilde{d} and d based on document topic distribution. The difference between them is the final similarity between \tilde{d} and d . After calculating all similarity, we assign \tilde{d} to the author of document d with the largest similarity value. In this way, both authors and documents play important roles in authorship attribution, which is never considered before to our knowledge. The classification process is shown in Algorithm 2.

Algorithm 2. Author assignment for test documents

Input:

- 1: A_d : the author of training documents set
- 2: \mathbf{D} : the training documents matrix;
- 3: $\tilde{\mathbf{D}}$: the test documents matrix;
- 4: $\theta^{(A)}$: the author topic distribution;
- 5: $\theta^{(D)}$: the document topic distribution in the training documents \mathbf{D} ;
- 6: $\tilde{\theta}^{(D)}$: the document topic distribution in the test documents $\tilde{\mathbf{D}}$;
- 7: $\phi^{(A)}$: the author topic word distribution;
- 8: D : the corpus matrix;

Output: *result*: authors of test documents $\tilde{\mathbf{D}}$;

```

9: procedure AUTHOR ASSIGNMENT FOR TEST DOCUMENTS
10:   for all test documents  $\tilde{d}_i \in [1, \tilde{D}]$  do
11:     for all training documents  $d_j \in [1, D]$  do
12:        $similarity_{i,j} = \text{Similarity}(\tilde{d}_i, d_j)$  by equation (13)
13:       if  $similarity_{i,j} > \text{currentMaxSimilarity}$  then
14:          $\text{currentMaxSimilarity} = similarity_{i,j}$ ;
15:          $\text{currentAuthor} = a_{d_j}$ ;
16:       end if
17:     end for
18:      $a_{\tilde{d}_i} = \text{currentAuthor}$ ;
19:   end for
20: end procedure

```

3.4 ADT vs DADT

ADT seems a little similar to DADT because both of them build the topic model for the corpus at the document level and the author level, but there are two key differences between them.

Firstly, we treat every word of the corpus equally at both two levels in ADT, which is more suitable for short texts. When we apply DADT to short texts, the words of the corpus are divided into two disjoint sets. In other words, each word of the corpus has an author topic and a document topic in our model, but it only has an author topic or a document topic in DADT.

Secondly, considering the effect of both authors and documents, we propose a new classification algorithm to find the most probable author of the anonymous texts. In this way, both authors and documents play important roles in authorship attribution for short texts.

Table 1. Detail properties of the datasets.

Dataset	Pan'11	Blog
Authors	71	136
Avg.texts	143	61
Avg.words \pm stdev	39 \pm 33	57 \pm 52

4 Experiments

4.1 Datasets

We use two public datasets for the experiment evaluations. The first one is **Pan'11 emails** [8]: 11936 emails with 72 authors. The second one is **Blog** [9]: 678161 blogs with 19320 authors. We want to figure out the problem of authorship attribution for short texts. Hence we delete emails and blogs which have more than 1000 characters. Besides, considering the time and space constraints, we only select 136 prolific authors from blogs. After pre-processing, some statistics about the datasets are provided in Table 1.

A quick analysis of the two datasets shows that the blog dataset is noisier than the Pan'11 dataset. Compared to Pan'11, Blog has more authors but fewer documents. The average length of documents from the blog is also longer than the Pan'11 dataset. In addition, we remove one author's documents from Pan'11 because they all have more than 1000 characters. We use both datasets to evaluate our ADT model's stability of performance on different scales of the corpus. We indicate the significance using t-test, two-tailed, p-value < 0.05 .

4.2 Baselines

Character 4-gram is an effective feature for authorship attribution, but we find it only has a positive effect on SCAP [10], and FS [3]. In topic models, like AT, DADT [14] and our proposed method ADT, or SVM, compared with word feature, character 4-gram does not perform better but needs more time to train the topic models. Considering the cost of time, we model the corpus with character

4-gram features in SCAP, and FS, and we model the corpus with word features in topic models and SVM.

SCAP. We build the author profiles for each of the candidates from the corpus and calculate the Jaccard similarity between a given text and the profile texts of authors to find the most similar author.

FS. We test different values for the parameter of the feature sampling method k . We find when we randomly sample 40% features from the feature set and $k = 100$, the best performance is achieved.

SVM. SVM is widely used in text classification. We use linear SVM in a one-versus-all setup, as implemented in Weka [17].

AT. We calculate the probability of each test text for each author by the inferred Author-Topic model and return the most probable author. Compared to LDA-H [7], AT significantly performs better when the corpus has tens of authors [14].

DADT. DADT is our most important baseline for comparison. When we get an inferred DADT model, the model assumes that test texts are written by a new author, and use the given model to infer the author/document topic ratio and the document topic distribution. Then we calculate the probability of each author for test texts and return the most probable author.

Table 2. Accuracy with different topic values on ADT and DADT. The best performance is highlighted in bold.

K_A/K_D	40/10	90/10	140/10	190/10	240/10
Pan'11					
ADT	49.2%	54.0%	54.2%	54.2%	54.7%
DADT	45.2%	51.2%	51.2%	51.1%	51.3%
Blog					
ADT	36.9%	45.6%	48.5%	49.2%	48.9%
DADT	32.9%	41.1%	44.4%	45.7%	45.5%

4.3 Experiment Setup

In all experiments, ten-fold cross-validation is carried out on both two datasets. We first trained all the methods on the training set, tuned the parameter according to the results on the test set to get the best performance. We use classification accuracy which is the percentage of test texts that are assigned to the correct author for evaluating results.

To train the topic models, we use collapsed Gibbs sampling with a burn-in of 1000 iterations. In all of the experiments with topic models, we retain 100 of samples with the spacing of 1 iterations. According to Eqs. (7), (8), (9) and (10), we estimate the author topic distribution, document topic distribution,

author word distribution and document word distribution from training samples for our models. Then we average them to get stable parameter estimates for the models. In classification phase, we use a burn-in of 100 iterations and average the parameter estimates over the next 100 iterations to get the final document topic distribution for test texts. In addition, we set $\alpha^{(A)}$ and $\alpha^{(D)}$ to $\min\{0.1, 5/K_A\}$ and $\min\{0.1, 5/K_D\}$, $\beta^{(A)}$ and $\beta^{(D)}$ to 0.01 for getting the best performance.

4.4 Results

Table 2 shows the accuracy of ADT and DADT with different values of author topic K_A on Pan’11 and Blog. We find that when we increase the value of document topic K_D , it does not improve the performance of both models, so we set K_D to 10 in all experiments. Compared to Pan’11, Blog needs more author topics to obtain the relatively stable ability of classification for the models. The results indicate that the performance of ADT outperforms DADT with all different values of K_A .

Table 3. Accuracy with different models on two datasets. The best method is highlighted in bold.

Dataset	Pan’11	Blog
ADT	54.7%	49.2%
DADT	51.3%	45.7%
AT	47.8%	43.6%
SVM	47.0%	30.8%
FS	43.9%	43.4%
SCAP	22.9%	26.5%

The results of our model and other baselines on Pan’11 and Blog are shown in Table 3. Except for SCAP, all the methods yield relatively low accuracies on Blog, but the accuracy of FS only decreases a little on Blog. As we can see, three kinds of topic models achieve better performance on both two datasets than other methods. This is because topic model can handle the problem of data sparsity on the short texts. Compared to the state of the art, our ADT model obtains a 6.63% improvement on Pan’11 and a 7.66% improvement on Blog, which demonstrates the effectiveness and stability of ADT.

5 Conclusions

The main goal of this paper is to deal with the problem of authorship attribution for short texts. We propose a novel model ADT which combines Author-Topic model and traditional LDA to build the model for the corpus due to the irregular

usage and data sparsity of short texts. Compared to the traditional methods, ADT tries to make use of the effect of training texts on authorship attribution, and we propose a new classification algorithm which combines the influence of both authors and texts to calculate the similarity between texts for finding the most probable author of test texts instead of only aggregating texts into a lengthy pseudo-document. On both real-world datasets, our proposed method performs significantly better than the current state of the art. In the future, we would like to combine the time factor and topic model to improve the performance of authorship attribution for short texts.

Acknowledgements. This work is supported by the National Natural Science Foundation of China [grant number 61772289] and the Fundamental Research Funds for the Central Universities.





References

1. Stamatatos, E.: A survey of modern authorship attribution methods. *J. Assoc. Inf. Sci. Technol.* **60**(3), 538–556 (2009)
2. Abbasi, A., Chen, H.: Writeprints: a stylometric approach to identity-level identification and similarity detection in cyberspace. *ACM Trans. Inf. Syst.* **26**(2), 1–29 (2008)
3. Koppel, M., Schler, J., Argamon, S.: Authorship attribution in the wild. *Lang. Resour. Eval.* **45**(1), 83–94 (2011)
4. Azarbondyad, H., Dehghani, M., Marx, M., Kamps, J.: Time-aware authorship attribution for short text streams. In: *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 727–730 (2015)
5. Hofmann, T.: Probabilistic latent semantic indexing. In: *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 50–57 (1999)
6. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. *J. Mach. Learn. Res.* **3**, 993–1022 (2003)
7. Seroussi, Y., Zukerman, I., Bohnert, F.: Authorship attribution with latent Dirichlet allocation. In: *Fifteenth Conference on Computational Natural Language Learning*, pp. 181–189 (2011)
8. Argamon, S., Juola, P.: Overview of the international authorship identification competition at PAN-2011. In: Petras, V., Forner, P., Clough, P. (eds.) *Notebook Papers of CLEF 2011 Labs and Workshops*, Amsterdam, Netherlands, 19–22 September 2011
9. Schler, J., Koppel, M., Argamon, S., Pennebaker, J.W.: Effects of age and gender on blogging. *Front. Inf. Technol. Electron. Eng.* **274**(s 1–2), 199–205 (2006)
10. Frantzeskou, G., Stamatatos, E., Gritzalis, S., Chaski, C.E., Howald, B.S.: Identifying authorship by byte-level N-grams: the source code author profile (SCAP) method. *Int. J. Digit. Evid.* **6**(1), 1–18 (2007)
11. Koppel, M., Schler, J., Argamon, S., Messeri, E.: Authorship attribution with thousands of candidate authors. In: *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 659–660 (2006)

12. Sousa Silva, R., Laboreiro, G., Sarmiento, L., Grant, T., Oliveira, E., Maia, B.: ‘twazn me!!!;’ Automatic authorship analysis of micro-blogging messages. In: Muñoz, R., Montoyo, A., Métais, E. (eds.) NLDB 2011. LNCS, vol. 6716, pp. 161–168. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-22327-3_16
13. Schwartz, R., Tsur, O., Rappoport, A., Koppel, M.: Authorship attribution of micro-messages. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pp. 1880–1891 (2013)
14. Seroussi, Y., Bohnert, F., Zukerman, I.: Authorship attribution with author-aware topic models. In: Meeting of the Association for Computational Linguistics: Short Papers, pp. 264–269 (2012)
15. Yang, M., Zhu, D., Tang, Y., Wang, J.: Authorship attribution with topic drift model. In: AACL, pp. 5015–5016 (2017)
16. Griffiths, T.L., Steyvers, M.: Finding scientific topics. *Proc. Natl. Acad. Sci. U.S.A.* **101**(Suppl 1), 5228 (2004)
17. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA data mining software: an update. *ACM SIGKDD Explor. Newsl.* **11**(1), 10–18 (2009)



WalkToTopics: Inferring Topic Relations from a Feature Learning Perspective

Linan Gao¹ , Zeyu Wang²  , and Shanqing Guo^{1,3} 

¹ School of Software, Shandong University, Jinan 250100, China
linangao98@gmail.com, guoshanqing@sdu.edu.cn

² Taishan College, Shandong University, Jinan 250100, China
zywangx@gmail.com

³ Key Laboratory of Cryptologic Technology and Information Security, Ministry of Education, Shandong University, Jinan 250100, China

Abstract. The increasing number of documents is leading to more and more topics nowadays. Understanding the relations between different topics evolved in documents become more important and challenging for users. Although many topic models have been devoted to analyzing topics, the study of topics' potential relevances is still largely limited by various difficulties. Hence, we introduce WalkToTopics, an unsupervised topic mining and analysis model, for inferring potential relevances between different topics. Relying on an advanced feature learning technique to automatically summarize topic's neighborhood features, WalkToTopics can reveal latent relations between different topics. Compared to existing approaches, our model is able to predict the relationship between any two individual topics of documents, and it does not require any prior knowledge of the existing topics' relations and dictionaries. Moreover, WalkToTopics is a general model that also can work on exploring topic clusters or extracting sentiments, and can be applied to potential applications, such as ideas tracking and opinion summarization. Finally, we conducted two studies for common users and experts which both quantitatively and qualitatively demonstrate the effectiveness of WalkToTopics in helping users' understanding of hidden relevances between topics on social media.

Keywords: Topic relations · Feature learning
Random walk · Relevance prediction

1 Introduction

With the wide-spread development of information technology and its applications, huge ever-increasing amounts of text collections and documents have been published to the internet. Documents are employed for sharing sentiments, conveying messages, as well as recording transactions. Hence, gathering information from opinions and relations about specific topics around texts is important to

many users such as administration staffs, politicians, business entrepreneurs, et al., which can help them run neck and neck with the latent topics and make suitable considerations. However, to extract relevance among topics is somewhat difficult and requires time and elaborations.

Researchers have proposed approaches facilitating document analysis for helping users cope with relations for the rising number of topics. Previous studies [3, 6–8] have proposed many different topic models and demonstrated that document data can indeed work as a strong predictor for analyzing the sentiments implicated in the topics. In addition, some works [2, 5, 14] also focus on the consideration of the evolution involved in massive topics over time. Nonetheless, these approaches are mainly conducted with a sentiments classification perspective and based on a probability distribution model, i.e., models do not focus on revealing the relations of topics. Moreover, the generating process of a topic model notably impose several parameters needed to be set which have an impact on the output topics of the model, and the result may perform not so well if there is no prior understanding of the dictionary to be modeled [1].

Therefore, we introduce WalkToTopics, a novel unsupervised topic mining and analysis model based on feature learning and random walk process, for helping users better comprehending the potential relations around different topics. Beyond the model, a feature learning perspective for topic analysis and modeling is highlighted in this work. Moreover, we propose a feature learning approach, namely vector mapping model, for predicting the latent links among different topics.

The remainder of this paper is divided into five sections. In Sect. 2 we briefly reviewing previous representative works directly related to our work. Then we detailed describe how to formulate WalkToTopics in Sect. 3. Hereafter, we introduce two extensions for enhancing our model in Sect. 4. By then, We conduct two case studies and analysis the evaluation results in Sect. 5. Finally, we conclude this model and discuss the future work in Sect. 6. In summary, the main contributions of this paper include:

- a novel feature learning based topic analysis model, WalkToTopics, for inferring topic relations;
- the introduction of the random walk based vector mapping to better characterize the topics’ distance in feature learning; and
- conducting an empirical study and an expert validation for showing the effectiveness of WalkToTopics.

2 Related Work

In this section, we briefly revisit two categories of previous studies directly related to ours, including varying kinds of topic models and the graph-theoretic random walk stochastic process.

2.1 Topic Models

Topic model is one of the most useful techniques for texts' data mining. Latent Dirichlet Allocation (LDA) [3] is the most popular probabilistic generative form of topic models which has already been widely applied to various extensions. There are some notable examples that use auxiliary information such as the supervised Latent Dirichlet Allocation (SLDA) [7]. Blei and Lafferty [2] present the Dynamic Topic Model (DTM) approach based on the dynamic LDA for modeling time evolution of topics by applying a state space model on the natural parameters, it also can work on the variational approximations based on Kalman filters and wavelet regression. Different from DTM and its variations, the Topics over Time (ToT) model [14] formulates each topic with a Beta distribution over time to directly capture evolutions. Another famous LDA-based extension is Joint Sentiment Topic (JST) model [8] which aimed to detect sentiments and topics through employing a sentiment layer addons on LDA. Moreover, the Topic-Sentiment Mixture (TSM) model [6] performs well for revealing the potential topics of Weblogs via constructing a hidden Markov model. Despite these models can reveal some information of the sentiments or evolution for the topics, users still can not easily comprehend how related each pair of the topics are.

2.2 Random Walk

Random walk is a general probability theory model. This kind of stochastic process is conducted by a successive summation approach of independent, identically distributed random variables. For instance, one-dimensional random walk cases are always regarded as a Markov chain with a series of opposite integers as state space, regularly formed by $0, \pm 1, \pm 2$, and so on.

To better explain, we first denote a topics' graph as $G = (V, E)$, comprising the set of vertices V and the set of edges E . Assume there is a random walk processing on a graph G starting at a vertex v , the approach tends to a sequence of vertices comprising each vertex as a random neighbor iteratively, i.e. comparing v_1 with v , v_2 with v_1 , and so on. In other words, by regarding the random walk process as a Markov chain, its transition probability matrix P can be denoted as:

$$P(i, j) = \begin{cases} 1/d_i, & \text{if } ((i, j) \in E), \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

where d_i refers the degree of vertex i . Meanwhile, the state distribution also changes with the iterations increasing, the output of state distribution S after t times iterations can be formulated as $S_t = S_0 * P(t)$, in which S_0 is the initial state distribution and $P(t)$ denotes to the t -th transition probability matrix. From these representation, thus random walk approach can work on inferring the relations between each pair of vertices on the graph G .

3 Methodology

WalkToTopics is designed for inferring topics' relationship existed in a document collection. In this section, we provide an overview and introduce the detailed processing and organization of our WalkToTopics model.

3.1 Definition

In this section, we formally define the general terminologies of WalkToTopics model. The definitions of four basic geometry structures extensively applied in this paper are proposed as follows (Fig. 1 illustrates a simple instance):

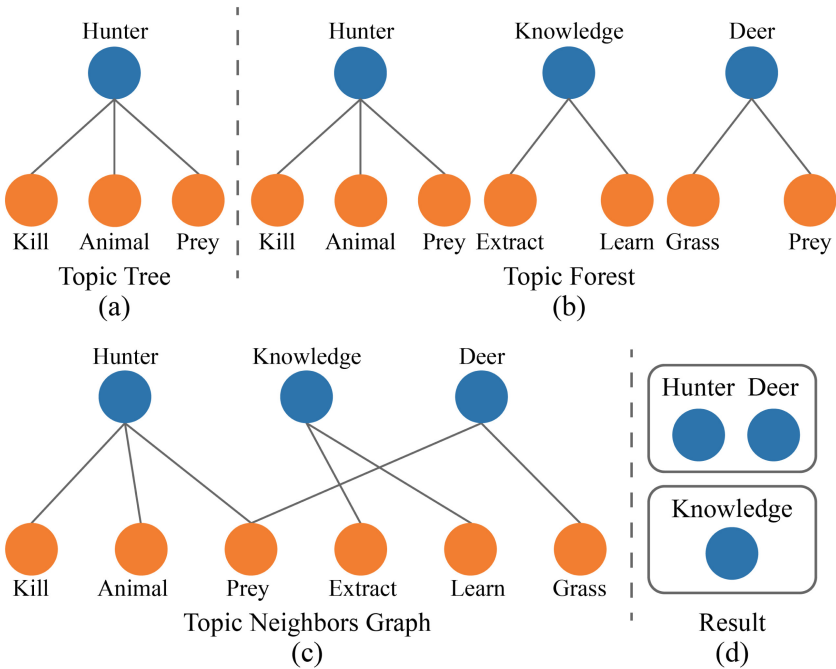


Fig. 1. A simple instance of these definitions, topics in blue, non-topic words in yellow. (a) shows a *Topic Tree* example (one topic three leaves). (b) is a *Topic Forest* with three topics. (c) indicates the *Topic Neighbors Graph* generated by the *Topic Forest* shown in (b). (d) presents the clustering result of the graph shown in (c). (Color figure online)

Definition 1 Topic Tree. A *Topic Tree* T in a set of words is a two-level structure that infers the relevant words of its topic, in which the root node refers the topic word, and the leaf nodes are other words that related to the topic. One *Topic Tree* can be constructed by merging some other *Topic Trees*.

Definition 2 Topic Forest. A *Topic Forest* T_F is a set of *Topic Trees* extracted from the same input document collections.

Definition 3 Topic Neighbors Graph. A *Topic Neighbors Graph* is a weighted bipartite graph $G = (V, L, E)$ conducted from the *Topic Forest* of the same document collection, where V refers the root nodes (i.e., topics) set of forests, L is the leaf nodes (i.e., words related to the topic) set of each topic in its corresponding *Topic Tree*, and $E \in V \times L$ contains all the edges between V and L .

Definition 4 Topic Clustering Graph. A *Topic Clustering Graph* is a weighted complete graph $G_C = (V, E)$, where V refers to the set of topics, E is the edges, and the weight of each edge is the similarity value between its two linked topics.

3.2 Initialization

Our first stage is to build up the basic geometry structures. For a document collection, WalkToTopics first remove the most common words that are unimportant and uninteresting, such as “the”, “that”, “thus”, and so on. Hereafter, for each sentence, a series of *Topic Trees* are constructed by regarding each of the remaining words into topics, while the others are treated as related words. Furthermore, the Porter Stemming Algorithm [11] is employed to combine similar words such as “sit” and “sitting”. Finally, a *Topic Forest* is built through merging *Topic Trees* with the same topic words.

Here, we consider a non-interested topic to be a word whose merging times smaller than ω . ω is defined by the following formula:

$$\omega = mt_{75} + 1.25(mt_{75} - mt_{25}) \quad (2)$$

where mt_{75} indicates the 75th percentile of all topics’ merging times and the expression in the parentheses is the *interquartile range* (IQR) of global merging times.

3.3 Tracking Neighbors with Random Walk

In the second stage, we formulated topics and its related words into a weighted bipartite graph $G = (V, L, E)$, where we define the weight of an edge (u, v) in E as the number of times that u and v appeared in the same sentence. The most common way to define the neighbors of topics is applying depth-first sampling (DFS) or breadth-first sampling (BFS). However, these results cannot reveal the similarities between different topics and the frequency information (edge weights) that two words appear together appropriately. Hence, the random walk stochastic process can better fit for our tasks.

The output is a random walk traces’ set D_R which composed by a series of single trace D_i . Hyper-parameters *times* and *length* indicate how many times

Algorithm 1. Random Walk to Neighbors

Input: A bipartite graph $G = (V, L, E)$
Output: Random walk traces D_R of G

```

1  $D_R \leftarrow \emptyset;$ 
2 for  $v \in V$  do
3   for  $i = 1$  to  $times$  do
4      $D_i \leftarrow v; n_{from} \leftarrow v;$ 
5     for  $j = 2$  to  $length$  do
6        $N_{neighbors} \leftarrow GetNeighbors(G);$  // neighbor set of  $n_{from}$ ;
7        $w_{tmp} \leftarrow GetWeight(n_{from}, n_{to});$  // topics' weight;
8        $n_{to} \leftarrow DFS(N_{neighbors}, w_{tmp});$ 
9        $D_i \leftarrow D_i \cup n_{to};$ 
10       $n_{from} \leftarrow n_{to};$ 
11    end
12     $D_R \leftarrow D_R \cup D_i;$ 
13  end
14 end

```

and how long should it walk during one cycle. As shown in Algorithm 1, we first find all neighbors for each input word (topic or non-topic word), and then compute the weight from topic to the non-topic word. The transition probability of random walk between two nodes is selected by the alias algorithm [13] as follows:

$$P(i = n_{from}, j = n_{to}) = \begin{cases} w_{i,j}/\dot{d}_i, & \text{if } (i \in V \& (i, j) \in E), \\ w_{j,i}/\dot{d}_j, & \text{if } (j \in V \& (j, i) \in E), \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

where \dot{d}_i refers to the corresponding weight of vertex i , i.e., the summarized weights of the edges connected to i . In the end, we obtain the random traces set D_R including $|V| \times times$ traces based on the transition probability. Hence the definite neighbors of a topic t , denoted by $N(t)$, are their adjacent nodes (i.e., precedent and after t) in the traces set D_R . It's also worth to note that these traces will be much more stable with $times$ and $length$ increasing which will make it more time-consuming, hence their values are set experimentally.

3.4 Vector Mapping Model

Thirdly, we build a vector mapping model in order to transform the random work traces for mapping each topics' correlated words into a continuous vector. This model is modified based on the skip-gram model of word2vec. Vector mapping model outputs one vector for each input topic which represents its relevant features of random walk traces. By modifying word2vec, vector mapping will input with fitful text collections and result in good embeddings, i.e. two topics having more similar neighbors from sentences will keep more closer in the vector space, which makes it suitable for our task.

Vector mapping model employs all the topics for predicting relevances. For a corpus of topics w and their neighbors $N(x)$, our goal is to maximize the conditional probabilities p . Hence its objective function can be expressed as follows:

$$\arg \max_{\theta \in \mathbb{R}^{|V \cup L| \times N_{dim}}} \prod_{x \in |V \cup L|} p(N(x)|x; \theta) \quad (4)$$

where θ refers to the mapped vectors of all topics in G , and N_{dim} presents the dimensions of corresponding vectors. Furthermore, by assuming that inferring neighborhoods of topics are independent for each other, Eq. 4 can be formalized into:

$$\arg \max_{\theta \in \mathbb{R}^{|V \cup L| \times N_{dim}}} \prod_{x \in |V \cup L|} \prod_{y \in N(x)} p(y|x; \theta) \quad (5)$$

Moreover, a normalized exponential function is conducted for modeling the conditional probability p :

$$p(y|x; \theta) = \frac{e^{\theta(x) \cdot \theta(y)}}{\sum_{z \in |V \cup L|} e^{\theta(x) \cdot \theta(z)}} \quad (6)$$

where $\theta(x) \in \mathbb{R}^{N_{dim}}$ is the mapped vector of x and $A \cdot B$ means the dot product of vectors.

By integrating Eq. 6 into Eq. 5 and applying logarithmic transformation, the object function will be turned into:

$$\arg \max_{\theta \in \mathbb{R}^{|V \cup L| \times N_{dim}}} \prod_{x \in |V \cup L|} \prod_{y \in N(x)} (\theta(x) \cdot \theta(y) - \ln \sum_{z \in |V \cup L|} e^{\theta(x) \cdot \theta(z)}) \quad (7)$$

which clearly reveals the close relations between two vertices with similar neighbors. Gradient descent method and hierarchical softmax function are adopted to optimize the objective function.

3.5 Inferring Relevances of Topics

Last but not least, we select a correlation coefficient similarity measurement to compare the feature vectors learned by vector mapping model for each pair of topics in G . The pairwise measuring formula is as follows:

$$s(\theta(x), \theta(y)) = \frac{(\theta(x) - \overline{\theta(x)}) \cdot (\theta(y) - \overline{\theta(y)})}{\|\theta(x) - \overline{\theta(x)}\|_2 \|\theta(y) - \overline{\theta(y)}\|_2} \quad (8)$$

where $\overline{\theta(x)}$ refers to the mean value of $\theta(x)$. According to the similarity value for topics, we can easily explore the potential relations around all vertices in G for each topics. And we can get a list of the most related words for each chosen topic by simplified sorting the topics over this value.

4 Extensions

In this section, we provide two probable extensions for WalkToTopics: topic clustering and sentiments extraction.

Topic Clustering. A force direct topic clustering method to find clusters behind these topics is employed by WalkToTopics. Through revisiting the results of above similarity measurement, we first construct a *Topic Clustering Graph* G_C , where the edge weights refer to the corresponding similarity value between two topics. Previous work [10] has shown that a unified prominent characterization could be solved by optimizing energy models of pairwise attraction and repulsion subsume Newman and Girvan’s modularity measure [9]. Hence, by emphasizing edge weights of G_C as the repulsive force between its two adjacent nodes, we can easily get an energy model and optimize this model to find better-clustered characterizations of topics.

Sentiments Extraction. The above explanation clearly indicates that our model can reveal potential relevances of topics. However, WalkToTopics still cannot work for sentiments analysis, which is also important for users. Formulating the topic clustering approach will result in a well-clustered graph in which the related topics are constellated together. Therefore, to generate high-level illustrated sentiments, we provide an extended sentiments extraction approach. Briefly speaking, this approach is to summarize topics of each obvious cluster into several sentimental words or phrases.

5 Evaluation

The evaluation process of WalkToTopics combined with two different approaches respectively. First, we conduct a small-scale quantitative empirical study with 30 participants. Hereafter, we discuss the results of a qualitative expert study with 5 professional users whom are regarded as experts in texts. The purpose of these studies is to compare WalkToTopics with state-of-the-art approaches and assess how much can our model help users better understand documents.

5.1 Empirical Study

In this study, our goal was to evaluate how well different topic analysis approaches perform in comprehending-oriented tasks. In detail, we compare the results of WalkToTopics with LDA [3] and TF-IDF [12] by employing scikit-learn python library.

Datasets. For preparation, we collect nine documents T_1 to T_9 including magazine articles and academic papers as experimental input datasets. Furthermore, it is worth to note that six documents (T_4 to T_9) are selected from English proficiency tests with several questions related to the texts. The documents are separated into two part: T_1 to T_3 for Task 1, and T_4 to T_9 for Task 2.

Tasks. By comprehensively exploring different available tasks and measuring how these tasks can be assessed, and how realistic they are, we finally decide to define the following two tasks for our study:

Task 1 - Emotion analysis. This approach is meant to exhaustively measure the emotional information of texts. This task encompasses with two subtasks: (i) summarize the global emotions using several words based on the three methods' results, and (ii) read the full article and modify the previous summarizations. The goal of this task was to allow users to easily and rapidly understand the main target of original articles (T_1 to T_3).

Task 2 - Semantic comprehension. Semantic interpretation would leave too much room for subjective interpretation, hence we proposed another object for users. The second task relates to testing how well can these techniques help understanding of documents in detail. Users were asked to answer the semantic-oriented questions (10 for each) of T_4 to T_9 only based on the output results of WalkToTopics, LDA, or TF-IDF without entire articles. This task was to test the users' comprehending for detailed information on original articles.

Note that the outputs of three methods include the results both for the entire article and for its each paragraph with different tags (e.g., D means the entire document, P_1 refers to the result of paragraph 1).

Hypotheses. Based on WalkToTopics' ability to infer the hidden relations of topics and words, we made following hypotheses before conducting the study:

H1: For both tasks, users will be more efficient with WalkToTopics than the other methods.

H2: For Task 1 (emotion analysis), user selected emotions will be more consistency with the help of WalkToTopics.

H3: For Task 2 (semantic comprehension), WalkToTopics will result in more accurate results, that is, with fewer errors.

Participants. We invited 30 participants (15 male, 15 female), most of whom are undergraduate students from the local university. Their ages ranged from 18 to 26 years (21 on average). All participants reported normal vision and none of them had prior experience with topic models and document analysis.

Procedure. We defined the following procedure for the study:

1. A 5-min introduction, including brief familiarization with the datasets and detailed explanation of tasks;
2. Task 1, for each document, users are randomly divided into three groups (10 users for each) with utilizing three methods;
3. Task 2, users are randomly divided into three groups for each document, too; and
4. feedback and a brief interview.

To avoid systematic bias through latent learning effects, we determined that (i) the orders of 2 and 3 are counterbalanced, and (ii) the user groups for each task and document are randomly generated and different.

Results. By following previous statistical analyses, which can overcome several biases and limitations of classical null hypothesis testing with p-values, we employed an estimation-oriented approach with confidence intervals (CI) [4].

The results of our quantitative measures are summarized in Fig. 2. We select mean values and errors as 95% CIs (lower values are better) for drawing the statistical boxplots of time shown in Fig. 2(a, c). These results of efficiency are consistent with the trends we predicted with $H1$. It shows a clear and strong trend that WalkToTopics performs more efficient than the other two methods.

In terms of accuracy, we counted the average consistency ratio (e.g., if 4 words selected at first, the ratio is 100% if the words didn't change finally, 75%

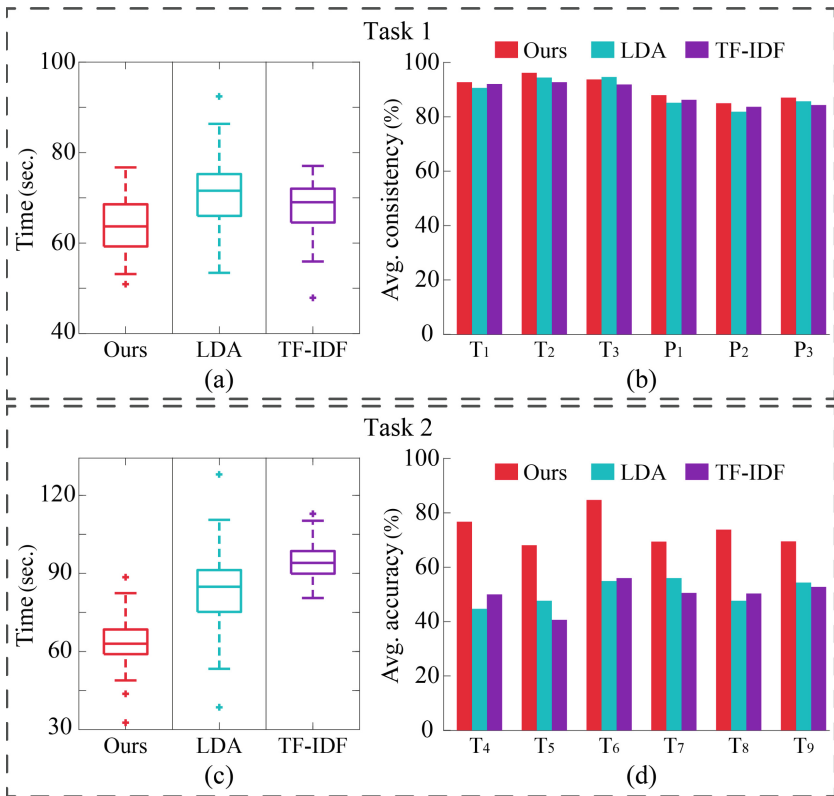


Fig. 2. Results of empirical study. P_i is meant to the paragraphs of T_i . (a, b) the boxplot of time users spent for each question and average consistency of Task 1. (c, d) the boxplot of time for the first summarization and average accuracy of Task 2. (Color figure online)

for changing 1 word of 4) on both articles and paragraphs for Task 1, see Fig. 2(b), and average accuracy ratio (i.e., the relative ratio between the number of right answered questions and all questions) for Task 2, see Fig. 2(d). (b) presents that all the three methods work well for keeping consistency, and WalkToTopics (red bar) performs slightly better than LDA (azure bar) and TF-IDF (purple bar) for most cases. The bar charts in (d) obviously reveal that the average accuracy of WalkToTopics is significantly greater than ones of the other methods. The results indicate that WalkToTopics was consistently better rated than the others by our participants which is similar to hypotheses *H2* and *H3*.

5.2 Expert Study

In the last stage, we wanted to analyze the importance of WalkToTopics through a more qualitative way. Therefore, we conducted case studies with professional users whom are our primary target audiences.

Study Design. We recruited 5 participants for this study. All participants engaged in some sort of text-oriented professions, such as journalist, litterateur, or official. The only task is to use WalkToTopics as a work utility according to their own taste, desire, and needs. They can choose no matter what genres of documents they’d like to. In contrast to the quantitative study, we offered the experts with entire spectrum of WalkToTopics’ functionality including extensions, we also provide the other two methods for better comparison.

Table 1. Subjective responses of experts

Questions	WalkToTopics	LDA	TF-IDF
It was easy to use	6.2	5.4	5.2
Its results were reasonable	5.6	4.8	5.0
It was effective	5.8	5.0	5.2
It was fun to employ	5.2	4.8	4.4
It simplified my work	6.2	5.0	4.6
I like to use this	6.0	5.2	4.2

Results. We gathered the experts’ results and comments on WalkToTopics after a three-day trial. And we were happy to observe that all 5 participants engaged in a creative utilize process and came up with inspiring and pivotal comments. By factorizing their feedbacks to 7-point Likert scale, we organized a questionnaire tabular ranging from strong disagreement to strong agreement as shown in Table 1. Hence, in the experts’ point of view, WalkToTopics is the most simple and effective one for helping on their works. We also asked them about their favorite functions, while 3 selected topic relation inferring and 2 preferred topic clustering extension.

6 Conclusion and Future Work

In this paper, we introduced the WalkToTopics model, a novel solution which can explore relations of topics. First, WalkToTopics embodies a random walk stochastic process by formulating the input documents into geometry graph. Hereafter, based on a vector mapping model, WalkToTopics can extract hidden relevances between topics. In addition, we presented two extensions for our approach: topic clustering and sentiments extraction. Finally, through conducting two empirical studies, we found that WalkToTopics exceeds some previous start-of-the-art approaches and performs well as a general utility which could improve work efficiency for text-oriented professions.

For future work, we plan to evaluate our model with standard benchmarks, find out the performance on larger datasets and study the difference with the latent cluster model (e.g., finite mixture model). Furthermore, it would also be interesting to combine our vector mapping processing with spotlight fundamental techniques such as LDA.

Acknowledgement. We thank all the anonymous reviewers for their insightful comments. This work is partially supported by National Natural Science Foundation of China (91546203), the Key Science Technology Project of Shandong Province (2015GGX101046), the Shandong Provincial Natural Science Foundation (ZR2014FM020), Major Scientific and Technological Innovation Projects of Shandong Province, China (No. 2017CXGC0704) and Fundamental Research Fund of Shandong Academy of Sciences (No. 2018:12-16).

References

1. Blair, S.J., Bi, Y., Mulvenna, M.D.: Increasing topic coherence by aggregating topic models. In: Lehner, F., Fteimi, N. (eds.) KSEM 2016. LNCS, vol. 9983, pp. 69–81. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-47650-6_6
2. Blei, D.M., Lafferty, J.D.: Dynamic topic models. In: Proceedings of the 23rd International Conference on Machine Learning, pp. 113–120. ACM (2006)
3. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. *J. Mach. Learn. Res.* **3**(Jan), 993–1022 (2003)
4. Cumming, G.: *Understanding the New Statistics: Effect Sizes, Confidence Intervals, and Meta-Analysis*. Routledge, Abingdon (2013)
5. Hu, Y., Xu, X., Li, L.: Analyzing topic-sentiment and topic evolution over time from social media. In: Lehner, F., Fteimi, N. (eds.) KSEM 2016. LNCS, vol. 9983, pp. 97–109. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-47650-6_8
6. Lin, C., He, Y., Everson, R., Ruger, S.: Weakly supervised joint sentiment-topic detection from text. *IEEE Trans. Knowl. Data Eng.* **24**(6), 1134–1145 (2012)
7. Mcauliffe, J.D., Blei, D.M.: Supervised topic models. In: *Advances in Neural Information Processing Systems*, pp. 121–128 (2008)
8. Mei, Q., Ling, X., Wondra, M., Su, H., Zhai, C.: Topic sentiment mixture: modeling facets and opinions in weblogs. In: Proceedings of the 16th international conference on World Wide Web, pp. 171–180. ACM (2007)
9. Newman, M.E., Girvan, M.: Finding and evaluating community structure in networks. *Phys. Rev. E* **69**(2), 026113 (2004)

10. Noack, A.: Modularity clustering is force-directed layout. *Phys. Rev. E* **79**(2), 026102 (2009)
11. Porter, M.: An algorithm for suffix stripping. *Program* **40**(3), 211–218 (1980)
12. Salton, G., McGill, M.J.: *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., New York (1983)
13. Walker, A.J.: An efficient method for generating discrete random variables with general distributions. *ACM Trans. Math. Softw.* **3**(3), 253–256 (1977)
14. Wang, X., McCallum, A.: Topics over time: a non-Markov continuous-time model of topical trends. In: *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 424–433. ACM (2006)



Distant Domain Adaptation for Text Classification

Zhenlong Zhu, Yuhua Li, Ruixuan Li^(✉), and Xiwu Gu

School of Computer Science and Technology,
Huazhong University of Science and Technology, Wuhan, China
{zzlong, idcliyuhua, rxli, guxiwu}@hust.edu.cn

Abstract. Text classification becomes a hot topic nowadays. In reality, the training data and the test data may come from different distributions, which causes the problem of domain adaptation. In this paper, we study a novel learning problem: Distant Domain Adaptation for Text classification (DDAT). In DDAT, the target domain can be very different from the source domain, where the traditional transfer learning methods do not work well because they assume that the source and target domains are similar. To solve this issue we propose a Selective Domain Adaptation Algorithm (SDAA). SDAA iteratively selects reliable instances from the source and intermediate domain to bridge the source and target domains. Extensive experiments show that SDAA has state-of-the-art classification accuracies on the test datasets.

Keywords: Text classification · Distant domain adaptation
Transfer learning

1 Introduction

Text classification now has many application scenarios such as social network mining and recommendations. However, most of the classification methods assume that training set and testing set conform to the independent and identical distribution hypothesis. In the real world, training set and test set likely come from different distributions, which creates the problem of domain adaptation.

Now there are many transfer learning algorithms that can partly solve the problem of domain adaptation. However, these algorithms have some limitations. They assume the source domain and target domain are different but related, which can be in the form of related instances, features or models. For two distant domains where no direct relation can be found, these transfer learning algorithms cannot achieve satisfactory results, and could lead to the ‘negative transfer’ [9] phenomena.

Tan’s work TTL [12] and SLA [13] aim to solve the problem of distant domain adaptation. TTL introduces intermediate domain to bridge the knowledge transfer between the source and target domains. TTL assumes that there must be an intermediate domain that all the data in it are helpful, which is not consistent

with the real situation. In many cases, this algorithm may not work well. SLA selects multiple subsets of instances from a mixture of intermediate domains as a bridge with the help of deep learning method. The depth network has specificity, and SLA’s image classification network does not apply to text classification. In this paper, we propose Distant Domain Adaptation for Text classification (DDAT) problem. In DDAT problem, the source domain can be very different from the target domain. Inspired by Tan’s work, we aim to bridge the source and target domains with the help of the intermediate domain.

In this paper, we aim to transfer knowledge between distant domains by selecting useful instances from the source and intermediate domains as a bridge to solve the DDAT problem. We use the manifold information to model the target data, which can capture the geometric features of the target data. The data in the source or intermediate domain which is related to the target data will be selected into the source training data. Then we transfer knowledge between source training data and target data by TJM [7]. The learning process of Selective Domain Adaptation Algorithm (SDAA) is an iterative process, which selects useful data from the intermediate and source domains to form source training data, and gradually modifies the target model until some stop criteria are reached. In our work, the *manifold loss* is a stop criterion for SDAA.

The contributions of this paper are summarized as follows:

1. In order to solve the limitation of traditional transfer learning methods in text classification, we propose the Distant Domain Adaptation for Text classification (DDAT) problem that the source domain is very different from the target domain.
2. To solve the DDAT problem, we propose Selective Domain Adaptation Algorithm (SDAA). In SDAA, the data related to the target domain are selected from the source and intermediate domains as the source training data. Considering the conditional distribution of the data, we use the *manifold loss* as our optimization goal to control the iteration of the SDAA.
3. Extensive experiments are conducted on two text datasets, and our method has obtained the best results compared with the eight baseline methods which indicates that our method is the state-of-the-art.

The rest of the paper is organized as follows. In Sect. 2, we discuss previous works related to our work and highlight their differences. The description of Distant Domain Adaptation for Text classification problem is given in Sect. 3. In Sect. 4, we present our proposed method. Experimental results and discussions are presented in Sect. 5 and finally we draw the conclusion in Sect. 6.

2 Related Work

In this section, we discuss previous works which are related to our method and analyze their differences.

Traditional transfer learning methods can be roughly divided into two categories: feature matching and instance reweighting. The feature matching method

is designed to reduce the distribution difference by learning a new feature representation [4, 6–8]. The instance reweighting method is designed to reduce the distribution difference by selecting relevant data from the source domain to help the learning in the target domain [3, 5]. However, due to their assumption that the source domain and the target domain are conceptually close, these approaches cannot handle the DDAT problem.

Tan has work on the problem of distant domain adaptation [12, 13]. The transitive transfer learning (TTL) [12] and the selective learning algorithm (SLA) [13] learn from the target domain with the help of the source domain and intermediate domains. TTL assumes that there exists an intermediate domain which can bridge source domain and target domain. However, in most cases, it seems difficult for us to find an useful domain. The selective learning algorithm (SLA) selects usefully unlabeled data gradually from intermediate domains to bridge the source and target domains with supervised autoencoder or supervised convolutional autoencoder as a base model. SLA also has some limitations that makes it difficult to solve our problem. First, SLA is a deep learning method for images classification which cannot handle the text classification. Second, in his experiments, the negative class in the source and target domains have the same distribution that may not be reasonable for the real world. In reality the negative class are more likely to come from different distribution.

There are also some transfer learning methods based on deep networks [11, 14]. Due to the features used in our work are not deep features, we will not discuss them.

Table 1. Notations and descriptions used in this paper.

Notation	Description	Notation	Description
S, T, I	Source/target/intermediate domain	Y_s	Source labels
T_a, T_b	Labeled/unlabeled target domain	Y_{ta}	Labeled target labels
X_s, X_t, X_I	Source/target/intermediate data	n_s, n_t, n_i	Number of source/target/intermediate data
X_{train}	Source training data	n_{ta}, n_{tb}	Number of labeled/unlabeled target data

3 Problem Definition

We first give the definitions of notations and concepts of DDAT problem, some of which have been defined in [7, 13]. For clarity, the frequently used notations are summarized in Table 1.

Definition 1 (Domain). A domain D is defined as an m -dimensional feature space χ and a marginal probability distribution $P(x)$, $D = \{x, P(\chi)\}$ with $x \in \chi$.

Definition 2 (Task). For domain D , a task T is composed of a C -cardinality label set γ and a classifier $f(x)$, $T = \{\chi, f(x)\}$, where $f(x) = Q(y|x)$ which can be interpreted as the class conditional probability distribution for each input sample x .

Definition 3 (DDAT problem). Given source labeled domain $S = \{(x_s^i, y_s^i)\}_{i=1}^{n_s}$, target labeled domain $T_a = \{(x_{ta}^i, y_{ta}^i)\}_{i=1}^{n_{ta}}$ where n_{ta} is not enough to learn an accurate classifier for the target domain, target unlabeled domain $T_b = \{x_{tb}^i\}_{i=1}^{n_{tb}}$ and unlabeled intermediate domain $I = \{x_I^i\}_{i=1}^{n_I}$. To keep the same setting for most of the transfer learning papers, we suppose the classification problems in the source domain and the target domain are both binary. In the problem, we have $\chi_S = \chi_T, \gamma_S = \gamma_T, P(x_s) \neq P(x_t) \neq P(x_I)$ and $Q(y_s|x_s) \neq Q(y_t|x_t)$.

The definition of the DDAT problem is similar to the DDTL problem [13], but there are also some differences between them. In their work, negative data from the source and target domains come from the same distribution which is not consistent with our problem.

Due to the large distribution gap between S and T , it may cause a substantial performance loss in target domain by directly transferring knowledge between them. The goal of our work is to bridge S and T with the help of the unlabeled intermediate domain I .

4 The Selective Domain Adaptation Algorithm

To solve the DDAT problem, we propose SDAA. First, SDAA selects useful data in the source domain and intermediate domain as the source training data. Second, it transfers the knowledge of the source training data to the target domain data. Then, we iterate over these two steps until the stop conditions are met.

4.1 Instance Selection

We choose the source training data whose structure is close to the target domain, and whose predicted label has high confidence. We use $u_s^i, u_t^i \subseteq \{0, 1\}$ to indicate whether we choose x_s^i, x_t^i to attend the source training.

First, we give *pseudo* labels $\{y_{pt}^i\}_{i=1}^{n_t}$ to target data, which are predicted by a classifier trained on the labeled data in the target domain. Second, according to the *pseudo* label, calculate the *manifold distance* of each data point of each class, which is calculated as Eq. 1.

$$dist_{t_c}^i = \sum_{j \in N_p^{t_c}(x_c^i)} \cos(x_c^i, x_c^j) \quad (1)$$

where x_c^i is the i th data point in the class c given by the *pseudo* label in the target domain, and $N_p^{t_c}(x_c^i)$ is the set of p -nearest neighbors of point x_c^i in class c . According to these *manifold distance*, we fit the classes into Gaussian models which are represented by $\{N(\mu^c, \sigma^c)\}_{c=1}^C$, and each Gaussian model represents the unique geometry of the class. Then, we calculate the *manifold distance* of each data point in the source and intermediate domains. The *manifold distance*

of the i th data point in the source and intermediate domains are represented by $dist_s^i$ and $dist_I^i$.

Last, a classifier $f_c(\cdot)$ is a classification function to output classification probabilities which is trained using the *pseudo* label of the target domain to classify the source and intermediate data, let $\{y_{p_s}^i\}_{i=1}^{n_s}$ and $\{y_{p_I}^i\}_{i=1}^{n_I}$ be the *pseudo* label of the source and intermediate data. We calculate u_s^i, u_I^i as Eq. 2.

$$u_s^i = \begin{cases} 1, & \text{if } y_{p_s}^i == y_s^i \text{ and } f_c(x_s^i) \geq \lambda_1 \text{ and } P(dist_s^i | \mu^{y_s^i}, \sigma^{y_s^i}) \geq \lambda_2 \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

$$u_I^i = \begin{cases} 1, & \text{if } f_c(x_I^i) \geq \lambda_1 \text{ and } P(dist_I^i | \mu^{y_I^i}, \sigma^{y_I^i}) \geq \lambda_2 \\ 0, & \text{otherwise} \end{cases}$$

where λ_1, λ_2 are the threshold parameters, which determine whether the data in the source and intermediate domains can be selected as the source training data.

4.2 Transfer Knowledge

Now, there are many excellent transfer learning algorithms aiming to jointly adapt both the marginal distribution and conditional distribution. They hold the assumption that the source domain and the target domain are similar and the *pseudo* labels in the target domain are close to its real labels. In DDAT problem, the target domain is very different from the source domain and the source training data depends on the target domain according to the instance selection method. It is unwise to consider conditional distribution, due to the distribution gap between source domain and target domain.

Here, we use TJM [7] to transfer knowledge from the source training data to the target domain. As an extension of TCA [8], TJM imposes the $l_{2,1}$ -norm structured sparsity regularizer on the source data, which reduces the domain difference by jointly matching the features and reweighting the instances across domains in a principled dimensionality reduction procedure. Through TJM, we can obtain the new feature matrix $X_{newsourc}$, X_{newtar} for the source data and target data, and based on the $X_{newsourc}$ and labeled data in X_{newtar} a classifier is trained to give *pseudo* labels to target data.

4.3 Iterative Refinement

According to the *manifold assumption* [1], if two points $x_s, x_t \in X$ are close in the intrinsic geometry of the marginal distributions $P_s(x_s)$ and $P_t(x_t)$, then the conditional distributions $Q_s(y_s|x_s)$ and $Q_t(y_t|x_t)$ are similar. Under geodesic smoothness, the *manifold loss* of the target domain is computed as Eq. 3.

$$M_f = \sum_{i,j=1}^{n_t} (y_{p_t}^i - y_{p_t}^j)^2 W_{ij} = \sum_{i,j=1}^{n_t} y_{p_t}^i L_{ij} y_{p_t}^j \quad (3)$$

where \mathbf{W} is the graph affinity matrix, and \mathbf{L} is the normalized graph Laplacian matrix. \mathbf{W} is defined as Eq. 4.

$$w_t^i = \begin{cases} \cos(x_t^i, x_t^j), & \text{if } x_t^j \in N_p^t(x_t^i) \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

where $N_p^t(x_t^i)$ is the set of p -nearest neighbors of point x_t^i in target domain. \mathbf{L} is computed as $\mathbf{L} = \mathbf{I} - \mathbf{D}^{-1/2}\mathbf{W}\mathbf{D}^{-1/2}$, where \mathbf{D} is a diagonal matrix with each item $D_{ii} = \sum_{i=1}^n W_{ii}$.

In this paper, we assume that the better the target domain classification, the smaller the *manifold loss*. We iterate through the first two steps until the *manifold loss* increases or reaches the upper limit of the number of iterations T , and the *pseudo* label for the target domain in each iteration is given by the previous round algorithm.

The complete learning algorithm is summarized in Algorithm 1.

Algorithm 1. SDAA: selective learning algorithm

Input: Data and labels: $X_s, X_{ta}, X_{tb}, X_I, Y_s, Y_{ta}$; number of iterations: T ;
nearest neighbor’s number: p ; threshold parameter: λ_1, λ_2

Output: adaptive classifier f

- 1 **Initialization:** set $t = 1$, train a classifier f_1 based on X_{ta}, Y_{ta} , give pseudo labels y_{p_t} to the X_{ta}, X_{tb} based on the f_1 , and calculate manifold loss M_{f_1} via Eq. 3, and set $M_{f_2} = INF, f = f_1$
 - 2 **repeat**
 - 3 **step1:** exchange M_{f_1} and M_{f_2} , set $f = f_1, t = t + 1$.
 - 4 **step2:** calculate u_s, u_I via Eq. 2, and select source training data X_{train} from the source and intermediate domains.
 - 5 **step3:** obtained the new feature matrix $X_{newsourse}, X_{newta}, X_{newtb}$ though TJM.
 - 6 **step4:** train a classifier f_1 based on $X_{newsourse}, X_{newta}, Y_s, Y_{ta}$.
 - 7 **step5:** give pseudo labels Y_{p_t} to X_{newta}, X_{newtb} based on f_1 .
 - 8 **step6:** calculate manifold loss M_{f_1} via Eq. 3.
 - 9 **until** $M_{f_2} < M_{f_1}$ or $T < t$
 - 10 **Return:** adaptive classifier f
-

5 Experiments

5.1 Data Preparation

In our experiments, there is only a weak connection between the source and target domains, and the labeled data in the target domain cannot provide much information. We obtain 101 datasets. The 101 distant-domain datasets are generated from 20-Newsgrroups and Multi-Domain Sentiment, which are the two benchmark text corpora widely used for evaluating transfer learning algorithms.

Table 2. 20-Newsgroups

domain	class	categories														average
source	positive	rec	talk	comp	sci	sci	talk	comp	rec	rec	talk	comp	comp	rec	sci	—
	negative	comp	comp	rec	rec	rec	rec	sci	sci	sci	sci	talk	talk	talk	talk	
target	positive	sci	sci	sci	comp	talk	sci	rec	comp	talk	rec	sci	rec	sci	rec	—
	negative	comp	comp	rec	rec	rec	rec	sci	sci	sci	sci	talk	talk	talk	talk	
\mathcal{A} -distance	—	1.038	1.02	1.320	1.3028	1.2736	1.3352	1.2872	1.3736	1.2860	1.2380	1.092	1.1924	1.1732	1.1472	1.2200

Table 3. Multi-Domain Sentiment

Domain	Class	Categories								Average
Source	Positive	electronics	kitchen	electronics	kitchen	book	dvd	book	dvd	—
	Negative	book	book	dvd	dvd	electronics	electronics	kitchen	kitchen	
Target	Positive	dvd	dvd	book	book	kitchen	kitchen	electronics	electronics	—
	Negative	book	book	dvd	dvd	electronics	electronics	kitchen	kitchen	
\mathcal{A} -distance	—	0.7670	0.7635	0.7820	0.7765	0.7155	0.7060	0.9990	0.9380	0.8059

20-Newsgroups¹ has approximately 20000 documents distributed evenly in 20 different subcategories. The corpus contains four top categories *comp*, *rec*, *sci* and *talk*. Each top category has four subcategories (e.g., top category *P*, four subcategories P_1, P_2, P_3, P_4). For each top category, we divide it into two subcategories (e.g., top category *P*, divided into P_a which consists of P_1 and P_2 , P_b which consists of P_3 and P_4). The four top categories are denoted by $comp_a, comp_b, rec_a, rec_b, sci_a, sci_b$ and $talk_a, talk_b$. We choose three top categories in an orderly fashion, which can produce $A_4^3 = 24$ sets of dataset groups. These three top categories are represented as A, B and C, the remaining category is D. For each dataset group, we randomly select a subcategory from B and C (e.g., B_a, C_a) as the positive class of the source and target domains, and randomly select a subcategory from A as the negative class for source domain and another subcategory in A as the negative class for target domain (e.g., A_a for source domain, A_b for target domain). A random sample of 30 samples is selected as the label sample in the target domain, and the rest of the subcategories are regarded as the intermediate domain. Under this strategy, for each dataset group, we can construct a total of $C_2^1 \cdot C_2^1 \cdot C_2^1 = 8$ sets of data. Finally, 24 dataset groups consisting of $24 \cdot 8 = 192$ datasets are generated.

We require the classifier trained on source domain to have more than 50% accuracy on the target domain, and the classifier trained on target labeled data to have less than 85% accuracy. Under this strategy, we obtain 14 dataset groups consisting of 87 datasets which are listed in Table 2, each document in the datasets is weighted by term frequency-inverse document frequency [10] ($TF - IDF$).

Multi-Domain Sentiment² has four top categories *books*, *dvd*, *electronics* and *kitchen*. Each top category has two subcategories *positive*, *negative*. We randomly select 1000 positive class data and 1000 negative class data for each top

¹ <http://people.csail.mit.edu/jrennie/20newsgroups>.

² <http://www.cs.jhu.edu/~mdredze/datasets/sentiment/>.

category. Using the similar strategy, we can conduct 8 dataset groups consisting of 14 datasets which are listed in Table 3. The only difference between the two strategies is that the positive and negative class of our domain must have different emotions.

5.2 Domain Distance

In our paper, we estimate the distribution difference of the source and target domains by \mathcal{A} -distance [2, 12]. From the definition of distance, the smaller the distance, the greater the difference between the two domains. We assign the source data positive labels, and target data negative ones, then select 100 data in the source and target domains and labeled them accordingly, and train a KNN classifier on these data to classify the source and target domains data, finally calculate the \mathcal{A} -distance as Eq. 5.

$$\text{dist}_{\mathcal{A}}(D_a, D_b) = 2(1 - \text{error}(h|D_a, D_b)) \quad (5)$$

From Tables 2, 3 and 4, we can see that the average \mathcal{A} -distance between the distant domains of **20-News** is 1.2200, and between the similar domains is 1.0704, while the average \mathcal{A} -distance between the distant domains of **sentiment** is 0.8059, and between the similar domains is 0.2713. From the results, we can obtain the conclusion that our source domains are far from the target domains.

Table 4. \mathcal{A} -distance of similar domains

—	Dataset						Average
20-news	comp vs rec	comp vs sci	comp vs talk	rec vs sci	rec vs talk	sci vs talk	—
\mathcal{A} -distance	1.1280	0.8956	0.8712	1.2980	1.1960	1.0336	1.0704
Sentiment	book vs dvd	book vs kitchen	electronics vs kitchen	electronics vs dvd	electronics vs book	dvd vs kitchen	—
\mathcal{A} -distance	0.2100	0.3440	0.2410	0.2480	0.2210	0.3640	0.2713

5.3 Baseline Methods

We compare SDAA with eight state-of-the-art supervised and transfer learning methods of text classification, excluding only the work based on the deep network, considering the fact that we do not use the deep features. They are:

- (1) 1-Nearest Neighbor Classifier trained on the source domain (NN1).
- (2) 1-Nearest Neighbor Classifier trained on the target domain labeled samples (NN2).
- (3) Transfer Component Analysis (TCA) [8] + NN.
- (4) Joint Domain Adaptation (JDA) [6] + NN.
- (5) Transfer Joint Matching (TJM) [7] + NN.
- (6) Boosting for Transfer Learning (TrAdaBoost) [3] + The decision tree.
- (7) Geodesic Flow Kernel (GFK) [4] + NN.
- (8) SDAAwi (SDAA without intermediate domain).

NN means 1-Nearest Neighbor Classifier, and which is chosen as the basic classifier since it does not require tuning cross-validation parameters.

5.4 Implementation Details

NN1 is trained on the source data; NN2 is trained on the target labeled data only; TCA, JDA and TJM are run on all data as dimensionality reduction step, then 1-Nearest Neighbor Classifier is trained on the source and target labeled data to classify the target data. For GFK, we first generate a series of intermediate spaces using all the source and target data to learn a new feature space, and then train a model in the space using all the source and target labeled data. TrAdaBoost is trained on the source and target data in a transductive way to directly induce domain-adaptive classifier. The training process of SDAAwi is the same as SDAA, except that SDAAwi does not use the intermediate domain data.

SDAAwi and SDAA involve four tunable parameters λ_1, λ_2, T and nearest neighbors p . Other hyper-parameters, such as the subspace dimensions and regularization parameters, are the same as the *TJM*. In our experiments we set $T = 4$ and $p = 5$ and (1) $\lambda_1 = 0.9, \lambda_2 = 0.91$ for **20-News**groups, (2) $\lambda_1 = 0.90, \lambda_2 = 0.80$ for **Multi-Domain Sentiment**. According to our experiment, when T value is 4, we can get good results and short running time.

In our experiments, *accuracy* on the test dataset is the evaluation measurement which is calculated as Eq. 6. It is widely used in literature, e.g. [7]

$$Accuracy = \frac{|x : x \in D_T \vee \hat{y}(x) == y(x)|}{|x : x \in D_T|} \quad (6)$$

where D_T is the target domain data treated as test data, $y(x)$ is the truth label and $\hat{y}(x)$ is the predicted label for a test data x .

Table 5. The experimental results in 20-news

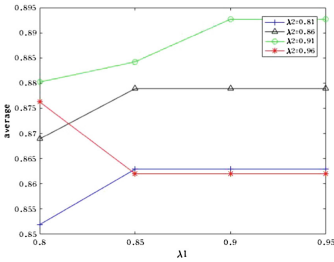
Data	KNN1	KNN2	TCA	JDA	TJM	TrAdaboost	GFK	SDAAwi	SDAA
rec comp sci comp	59.62%	79.74%	74.96%	74.27%	75.21%	50.49%	71.40%	81.23%	84.60%
talk comp sci comp	65.90%	79.74%	78.42%	77.53%	78.01%	50.49%	76.38%	83.73%	84.81%
comp sci rec sci	52.39%	83.86%	68.47%	67.83%	61.79%	51.02%	61.70%	89.92%	92.55%
rec sci comp sci	53.41%	77.23%	71.11%	68.83%	65.97%	49.51%	65.06%	78.76%	82.64%
rec sci talk sci	54.68%	75.53%	70.95%	70.12%	69.75%	50.09%	71.12%	81.32%	83.73%
talk sci rec sci	51.34%	82.90%	73.52%	77.48%	63.40%	50.09%	59.91%	86.87%	89.69%
comp rec sci rec	57.25%	82.71%	78.61%	81.85%	75.25%	49.85%	74.80%	85.56%	89.06%
sci rec comp rec	57.58%	84.62%	80.00%	73.64%	76.67%	49.40%	69.98%	89.96%	91.88%
sci rec talk rec	52.49%	83.64%	73.69%	70.04%	61.27%	41.44%	58.85%	92.90%	94.46%
talk rec sci rec	54.46%	82.40%	75.55%	76.54%	69.04%	49.87%	68.79%	85.72%	89.90%
comp talk sci talk	66.73%	79.27%	83.25%	82.38%	82.35%	55.04%	80.16%	90.80%	90.99%
comp talk rec talk	65.40%	81.63%	87.39%	81.40%	82.35%	55.14%	80.33%	91.68%	93.25%
sci talk rec talk	62.11%	81.63%	83.40%	79.64%	81.85%	55.14%	81.21%	92.21%	91.87%
rec talk sci talk	59.34%	79.27%	79.39%	78.53%	75.60%	55.04%	81.21%	88.94%	90.32%
Average	58.05%	81.01%	77.05%	75.72%	72.64%	50.28%	72.22%	87.11%	89.27%

5.5 Experimental Results and Discussion

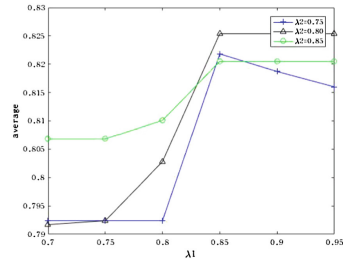
In this section, we compare our SDAA with the eight baseline methods in terms of classification accuracy.

Table 6. The experimental results in Multi-Domain Sentiment

Data	KNN1	KNN2	TCA	JDA	TJM	TrAdaboost	GFK	SDAAw1	SDAA
electronics book dvd book	55.62%	80.07%	81.08%	70.87%	76.65%	49.90%	67.80%	86.65%	84.73%
kitchen book dvd book	51.88%	80.07%	81.00%	73.35%	74.42%	48.98%	68.30%	82.95%	84.93%
electronics dvd book dvd	51.60%	82.50%	72.95%	66.35%	66.23%	48.33%	64.22%	81.80%	85.10%
kitchen dvd book dvd	52.35%	82.50%	77.02%	71.98%	67.95%	49.84%	63.82%	84.14%	84.53%
book electronics kitchen electronics	62.42%	81.20%	72.05%	78.33%	68.25%	49.82%	68.27%	79.08%	82.43%
dvd electronics kitchen electronics	57.20%	81.20%	70.80%	74.48%	66.90%	50.34%	65.22%	79.08%	83.30%
book kitchen electronics kitchen	60.70%	75.05%	69.15%	66.15%	61.35%	48.76%	62.00%	75.05%	77.65%
dvd kitchen electronics kitchen	63.60%	75.05%	70.10%	70.60%	64.85%	49.49%	62.35%	75.05%	77.65%
Average	56.92%	79.71%	74.27%	71.51%	68.33%	49.43%	65.25%	80.48%	82.54%



(a) parameters for 20-News.



(b) parameters for Sentiment.

Fig. 1. Parameters analysis.

20-Newsgroups: The classification accuracies of SDAA and the eight baselines are shown in Table 5. We observe that SDAA achieves much better performance than the baseline methods on most of the datasets. The average classification accuracy of SDAA on the 16 datasets is 89.27%, gaining a significant performance improvement of 8.26% compared to the best baseline NN2.

The accuracy variation w.r.t parameters λ_1 and λ_2 are shown in the left figure of Fig. 1. We can see that the accuracy increases first and then decreases with the increase of λ_2 . When λ_2 is between 0.81 and 0.91, the accuracy increases first and then remains unchanged with the increase of λ_1 . When λ_2 is equal to 0.96, the accuracy decreases and then remains unchanged with the increase of λ_1 . When λ_1 is equal to 0.90 and λ_2 is equal to 0.91, the accuracy is maximized.

Multi-Domain Sentiment: The classification accuracy of SDAA and the baseline methods on the 8 datasets are illustrated in Table 6. We observe that SDAA has the best performance. The average classification accuracy of SDAA on the 8 datasets is 82.54%, gaining a significant performance improvement of 2.83% compared to the best baseline NN2.

The accuracy variation w.r.t parameters λ_1 and λ_2 are shown in the right figure of Fig. 1. We can see that the accuracy increases first and then remains

unchanged with the increase of λ_1 when λ_2 is between 0.80 and 0.85. When λ_2 is equal to 0.75 the accuracy increases first and then decreases with the increase of λ_1 which is because the value of λ_2 is small, the selected source training data may not be similar in structure to the target domain data. When λ_1 is equal to 0.90, λ_2 is equal to 0.80, the accuracy is maximized.

From Fig. 1, we can draw a conclusion that with the increase of λ_1 and λ_2 , the accuracy increases first and then decreases or remains unchanged in most cases, and the high value of λ_2 may not play a good role in the accuracy. When λ_1 is about 0.9 and λ_2 is between 0.8 and 0.9, we can get good results.

From Tables 5 and 6, we can see that non-transfer methods cannot perform well on most datasets. NN1 does not perform well because there is too many difference between the source and target domains. For NN2, the target labeled data is too small to train a robust classifier.

Transfer learning methods such as TCA, JDA, TJM, TrAdaboost and GFK achieve worse performance than NN2 because the source domain and the target domain have huge distribution gap, which leads to ‘negative transfer’ [9]. GFK performs poorly because it continues to assume that the conditional distribution between the two domains is almost identical, which is inconsistent with our experimental setup. TCA is better than JDA in performance, because JDA takes into account the difference in the conditional distribution between domains and use the idea of EM algorithm to solve it. Due to the large differences between the source domain and target domain, the pseudo class centroids calculated by JDA may be far from the truth, so the JDA could fall into the local optimal value. In the case where the source and target domains are similar, TJM works better than TCA, which can be proven in article [6]. In this paper, the performance of TJM is much worse than TCA, because TJM needs iterative optimization parameters, but the two domains are very different, and cannot get a good initialization algorithm and iterative calculation will make the result much worse. TrAdaboost selects useful data from the source domain with the help of the labeled data in the target domain, but there is too little labeled data in the target domain to provide too much information, so TrAdaboost does a lot worse than expected.

SDAAwi achieves the best performance outside of SDAA, which proves the effectiveness of our selection strategy. But due to the large difference between the source domain and the target domain, the number of source data with similar structure of the target data is limited, which limits the performance of the SDAAwi. The result of SDAA is superior to SDAAwi, which proves that is correct to learn from the intermediate domain in DDAT problem.

6 Conclusion and Future Work

In this paper, we study the novel Distant Domain Adaptation for Text classification problem, where the source and target domains are distant. However, we can classify the target domain with the help of the intermediate domain. To solve the DDAT problem, we propose the selective domain adaptation algorithm (SDAA) to select unlabeled data from the intermediate domain to bridge the two distant domains. Experiments on the dataset show that SDAA is able to achieve state-of-the-art performance in terms of accuracy.

In the future, we will concentrate on the deep learning methods and study other text tasks, *e.g.*, entity recognition, within the setting of transfer learning.

Acknowledgements. This work is supported by the National Key Research and Development Program of China under grants 2016QY01W0202 and 2016YFB0800402, National Natural Science Foundation of China under grants 61572221, U1401258, 61433006 and 61502185. Guangxi High level innovation Team in Higher Education Institutions Innovation Team of ASEAN Digital Cloud Big Data Security and Mining Technology.

References

1. Belkin, M., Niyogi, P., Sindhvani, V.: Manifold regularization: a geometric framework for learning from labeled and unlabeled examples. *J. Mach. Learn. Res.* **7**, 2399–2434 (2006). [JMLR.org](http://jmlr.org)
2. Ben-David, S., Blitzer, J., Crammer, K., Pereira, F.: Analysis of representations for domain adaptation. In: *Advances in Neural Information Processing Systems*, pp. 137–144 (2007)
3. Dai, W., Yang, Q., Xue, G.-R., Yu, Y.: Boosting for transfer learning. In: *Proceedings of the 24th International Conference on Machine Learning*, pp. 193–200. ACM (2007)
4. Gong, B., Shi, Y., Sha, F., Grauman, K.: Geodesic flow kernel for unsupervised domain adaptation. In: *2012 IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pp. 2066–2073. IEEE (2012)
5. Huang, J., Gretton, A., Borgwardt, K.M., Schölkopf, B., Smola, A.J.: Correcting sample selection bias by unlabeled data. In: *Advances in Neural Information Processing Systems*, pp. 601–608 (2007)
6. Long, M., Wang, J., Ding, G., Sun, J., Yu, P.S.: Transfer feature learning with joint distribution adaptation. In: *2013 IEEE International Conference on Computer Vision, ICCV*, pp. 2200–2207. IEEE (2013)
7. Long, M., Wang, J., Ding, G., Sun, J., Yu, P.S.: Transfer joint matching for unsupervised domain adaptation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1410–1417 (2014)
8. Pan, S.J., Tsang, I.W., Kwok, J.T., Yang, Q.: Domain adaptation via transfer component analysis. *IEEE Trans. Neural Netw.* **22**(2), 199–210 (2011)
9. Pan, S.J., Yang, Q.: A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* **22**(10), 1345–1359 (2010)
10. Salton, G., Buckley, C.: *Term-Weighting Approaches in Automatic Text Retrieval*. Pergamon Press Inc., Oxford (1988)
11. Stewart, R., Ermon, S.: Label-free supervision of neural networks with physics and domain knowledge. In: *AAAI*, pp. 2576–2582 (2017)
12. Tan, B., Song, Y., Zhong, E., Yang, Q.: Transitive transfer learning. In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1155–1164. ACM (2015)
13. Tan, B., Zhang, Y., Pan, S.J., Yang, Q.: Distant domain transfer learning. In: *AAAI*, pp. 2604–2610 (2017)
14. Tzeng, E., Hoffman, J., Saenko, K., Darrell, T.: Adversarial discriminative domain adaptation. In: *Computer Vision and Pattern Recognition, CVPR*, vol. 1, p. 4 (2017)



Attention Aware Bidirectional Gated Recurrent Unit Based Framework for Sentiment Analysis

Zhengxi Tian¹ , Wenge Rong¹ , Libin Shi² , Jingshuang Liu¹ ,
and Zhang Xiong¹ 

¹ School of Computer Science and Engineering, Beihang University, Beijing, China
{tzx0505,w.rong,jingshuangl,xiongzg}@buaa.edu.cn

² Sino-French Engineer School, Beihang University, Beijing, China
libin.shi@buaa.edu.cn

Abstract. Sentiment analysis is an effective technique and widely employed to analyze sentiment polarity of reviews and comments on the Internet. A lot of advanced methods have been developed to solve this task. In this paper, we propose an attention aware bidirectional GRU (Bi-GRU) framework to classify the sentiment polarity from the aspects of sentential-sequence modeling and word-feature seizing. It is composed of a pre-attention Bi-GRU to incorporate the complicated interaction between words by sentence modeling, and an attention layer to capture the keywords for sentiment apprehension. Afterward, a post-attention GRU is added to imitate the function of decoder, aiming to extract the predicted features conditioned on the above parts. Experimental study on commonly used datasets has demonstrated the proposed framework's potential for sentiment classification.

Keywords: Sentiment analysis · Bidirectional GRU · Attention

1 Introduction

The advent of Internet and its convenience enable people to easily publish and share their own opinions and comments on purchased items, movies, restaurants and etc. in various social platforms [19]. It is important to analyze these online valuable information to understand customers' feedback and potential preference. Sentiment analysis has become an effective method to investigate the polarity of such online posted messages [26]. Identifying a user's emotional preferences from texts and classifying emotions into finite categories can be considered as a text classification problem [27].

Many methods have been developed in the literature for the task of sentiment analysis. For instance, the early proposed Single Label Learning (SLL) [23] and Multi-label Learning (MLL) [25] consider basic emotion as a class label, then compute the label scores to determine the sentiment of a review. Besides, emotion

lexicon can be also constructed to classify the sentiment polarity [23]. Recently, with the development of machine learning, particularly deep learning techniques, a lot of machine learning and deep learning based methods have been widely used in sentiment analysis [6].

Besides the needs of advanced network architecture, to better fulfill the emotion detection task, it is also necessary to understand how people perceive the sentiment polarity in reading a message. One important property of human perception is when they read a document, they not only need to focus on emotion words selectively to acquire specific information expressed by the text such as angry or satisfaction, but also combine information from different attention parts to build up a whole comprehension system [15]. Inspired by this, in this research we aim to analyze sentiment from the aspects of sentence modeling and word apprehension to identify the binary classification of reviews.

Considering that the sequence of a sentence is fundamental for comprehending by syntax dependencies, we first model the structure of a sentence to learn the underlying relations between words and their dependents. Recurrent Neural Network (RNN) [20] has proven its great capability in learning the underlying relationships between words in a sentence than the traditional models like support vector machines (SVM) [10] and Conditional Random Fields (CRF) [11]. Nonetheless, the vanishing and exploding problem [2] brought by RNN also limits its power. To address this limitation, many methods have been developed including advanced architecture such as Long Short-Term Memory (LSTM) [9] and Gated Recurrent Unit (GRU) [3]. As a simplification of LSTM, GRU has been widely used due to the more efficient computation while maintaining the advantages of LSTM.

Therefore, in this research, a pre-attention Bi-GRU is employed to incorporate information from words both preceding and following for constructing an initial comprehension in sentiment recognition. When the input sequences become longer, it is often difficult for LSTM and GRU to capture the significant part for global sentiment [22]. However, the specific words also make vital contribution to the sentiment analysis. Attention is an effective mechanism to capture the key information in a long sentence [1], which helps to classify emotion from the word-level aspect. Furthermore, we also added a post-attention GRU imitating the function of the decoder to extract the predicted features conditioned on the pre-attention Bi-GRU and the attention layer. Therefore our framework has the ability to analyze the sentiment polarity from both aspects of sentential-sequence modeling and word-feature seizing.

The rest of the paper is organized as follows: Sect. 2 introduces related work of our research. The detailed network architecture will be illustrated in Sect. 3. Section 4 will present extensive experiments and analysis. Section 5 will conclude this paper and point out possible future direction.

2 Background

Due to the massive amount of information on the Internet, sentiment analysis has developed several methods that are used in opinion mining, such as lexicon

based techniques [5], grammar rules and statistical methods [21]. In this case, most of the studies rely on bag-of-words or bag-of-n-grams to analyze texts from the perspective of tokenization, features selection, sentences parsing and documents classification. For instance, Naive Bayes (NB), SVM and other different classifiers [17] were used for detecting the polarity of movie reviews.

Recently with the rapid development of deep learning, artificial neural networks (ANNs) have been applied to various natural language processing (NLP) tasks. As an important ANN model, recurrent neural network (RNN) possesses the capacity of processing any length input without the need of increasing the model size. Besides, weights are shared across timesteps, so computation for one timestep can use information from many steps back. Therefore, RNN gives a better settlement in learning the underlying relationships between the words [20]. Nonetheless, recurrent computation is slow and the gradient can become very small or very large quickly [2], thus the locality assumption of gradient descent breaks down. It is what we called the vanishing gradient and exploding problems, which may cause information from many time steps in the past will not be taken into consideration when predicting the next word in language model or sentiment analysis tasks. To address this limitation, many methods have been developed including advanced architecture such as LSTM and GRU. Besides, some researchers [18] also use regularization term forces the error signal not to vanish. The traditional RNN process an input sequence from the starting to ending in a one-way direction.

A simplified LSTM [13] with gates and cell mechanism was introduced to address this limitation while maintaining the advantages of RNN. A stacked bidirectional LSTM proposed by Zhou and Xu [28] shows its better performance in handling longer sentences than traditional models. Based on their work, He [7] made improvements by using deep highway bidirectional LSTMs. With the advantage of getting annotation of each word not only from the preceding words but also from the following words, bidirectional RNNs have been widely used in NLP, especially in sentiment analysis area.

GRU combines forget gate and input gate from LSTM into update gate, which works with reset gate to control the information flow between the past and current input better [24]. With one gate less than LSTM, GRU is a simplification of LSTM, enabling it computes more efficiently. It's also the extension of the vanilla RNN unit, which is capable of alleviating the exploding and vanishing problem during training [16]. Therefore, we employed a pre-attention Bi-GRU to learn the word and syntax dependencies from the sentential-sequence level at the first step, in contrast to unidirectional model, our framework makes maximum utilization of interactive information from contexts.

Attention mechanism was first used in image classification [15] and improved by Bahdanau and Cho, who applied attention mechanism into neural machine translation (NMT). It allows the decoder to focus on certain parts of the source, that is to say, instead of considering information only from the last hidden state, all information related in the hidden states could be taken into account [1]. Attention also provides a shortcut to the distant state, which assists with

vanishing gradient problem as well. Attention mechanism is not only applied to NMT but also used in reading comprehension and put forward to solving the bottleneck issue, which helps enhance the accuracy of understanding sentiment of reviews greatly [8]. In this paper, we set the attention mechanism to focus on the most relevant and important part of input sequence processed from pre-attention Bi-GRU, which improve the performance of our framework.

3 Methodology

3.1 Sentential-Sequence Modeling by Pre-attention Bi-GRU

The whole architecture of proposed framework is depicted in Fig. 1¹. Specifically, we first formalize the notation used in this paper. We suppose that a sentence consists of m words $[w_i^1, w_i^2, \dots, w_i^m]$, where w_i denotes a specific word of a sentence. To represent the word, we embed each word into a one-hot-vector and then transform it to d dimensional matrix M through word embedding. As such for each word in sentence, we can get $w^k \in \mathbb{R}^d$ from $M^{v \times d}$, where k denotes the word index and v is the vocabulary size. A sentence in reviews can be represented by $x_i = [w_i^1, w_i^2, \dots, w_i^m]$.

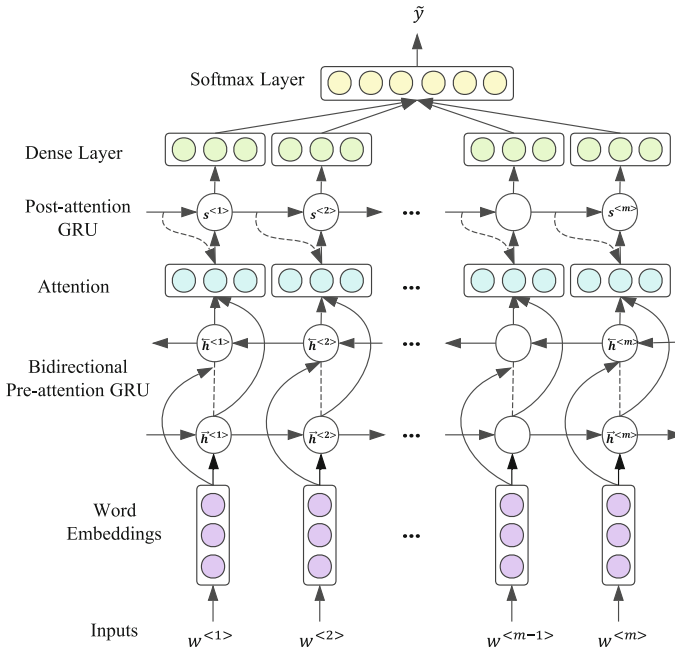


Fig. 1. The overall architecture of proposed framework

¹ The source code is available at <https://github.com/MercuryTian/Attention-Aware-Bi-directional-GRU-Based-Framework-for-Sentiment-Analysis.git>.

After initialization, we employ a pre-attention Bi-GRU to learn the underlying relationships between words, which comprehend sentiments from sentential-sequence level. The detailed architecture of Bi-GRU is shown at the bottom part of Fig. 1. GRU possesses gating units that modulate the flow of information inside the unit [4], and the bidirectional network structure combines information from words both preceding and following preferably. The ability to cope with the complicated interaction between the words and controlling the semantic flow between them is what the key function for sentential-sequence modeling.

The sentential-sequence modeling part consists of a forward and a backward sub-layer. The forward sub-layer receives the word embedding sequences starting from the beginning up to the end. The backward sub-layer does the opposite processing as the forward sub-layer. Formally, for the given input word embedding w^k , at each time step, its current and previous forward hidden state \vec{h}_t and \vec{h}_{t-1} , and its current and previous hidden backward state \overleftarrow{h}_t and \overleftarrow{h}_{t-1} in the pre-attention Bi-GRU are updated as:

$$z_t = \sigma(W_z w_t + U_z h_{t-1} + b_z) \quad (1)$$

$$r_t = \sigma(W_r w_t + U_r h_{t-1} + b_r) \quad (2)$$

$$\vec{h}_t = \tanh\left(W_{\vec{h}} w_t + U_{\vec{h}} \vec{h}_{t-1} + b_{\vec{h}}\right) \quad (3)$$

$$\overleftarrow{h}_t = \tanh\left(W_{\overleftarrow{h}} w_t + U_{\overleftarrow{h}} \overleftarrow{h}_{t+1} + b_{\overleftarrow{h}}\right) \quad (4)$$

$$\tilde{h} = g\left(V\left[\vec{h}_t : \overleftarrow{h}_t\right] + c\right) \quad (5)$$

$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot \tilde{h} \quad (6)$$

where z and r are update gate and reset gate respectively, which decide how much the unit updates or resets its activation, or content. σ means a sigmoid function and \odot stands for element-wise multiplication. $W_z, W_r, W_{\vec{h}}, W_{\overleftarrow{h}}, V \in \mathbb{R}^{d \times 2d}$ are the weighted matrices and $b_z, b_r, b_{\vec{h}}, b_{\overleftarrow{h}}, c \in \mathbb{R}^d$ are biases of Bi-GRU to be learned during training. The candidate memory cell \tilde{h} is computed by both forward hidden state and backward hidden state. h_t is the new state of memory cell. Then we get the hidden states $[h^1, h^2, \dots, h^m]$ as the final word representations for sentences.

3.2 Word-Feature Seizing by Attention Mechanism

After getting the last hidden states outputted from the first layer, we adopt the attention mechanism to assist the framework in judging sentiment polarity by focusing on the important information from word-feature level. The detailed architecture of our used attention mechanism is illustrated in Fig. 2.

As shown in the Fig. 2, the attention mechanism generates the attention distribution α_t^k at k^{th} time step by:

$$\alpha_t^k = \frac{\exp(e_t^k)}{\sum_{i=1}^m \exp(e_t^i)} \quad (7)$$

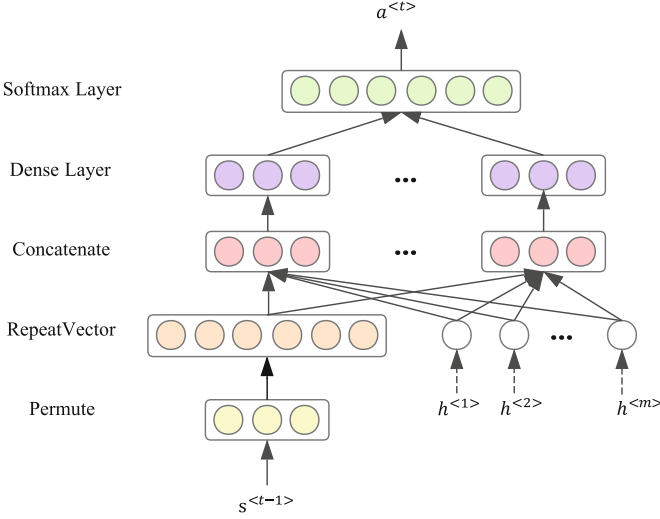


Fig. 2. The detailed structure of attention

where e_t^k is a score function for the memory cell of k^{th} time step. The score function e_t^k is defined as:

$$e_t^k = [s_{t-1}^T h_1, s_{t-1}^T h_2, \dots, s_{t-1}^T h_m] \quad (8)$$

where h_k is the hidden unit of pre-attention Bi-GRU and s_{t-1} is the memory cell of k^{th} time step in post-attention GRU. Next, we can get the attention output:

$$a_t = \sum_{k=1}^m \alpha_t^k h_k \quad (9)$$

3.3 Post-attention GRU

Afterward, a post-attention GRU follows by the attention layer to take the sentence level information into consideration by repeat reading the sentence deliberately like a human, imitating the function of the decoder to extract the predicted features. The main equations of post-attention GRU are similar with the Bi-GRU, except the definition of the candidate memory cell:

$$\tilde{h} = \tanh(W_{\tilde{h}_t} w_t + r_t \odot U_{\tilde{h}_t} h_{t-1}) \quad (10)$$

Finally, we send the outputs of post-attention GRU into a softmax classifier and get the prediction class label $\{positive, negative\}$ of the sentiment document.

4 Experimental Study

4.1 Experiment Configuration

Dataset. Our experiment on sentiment analysis are performed on the four public datasets, i.e. the IMDB large movie review², Customer Review³, Movie Review⁴ and SemEval 2014 Task 4⁵. The first three labeled datasets have 50,000, 4,530, 10,624 instances respectively and are composed of reviews in two polarities: *positive* and *negative*. The SemEval dataset contains two subsets: *Laptop* and *Restaurant*. The *Restaurant* dataset has 2,892 instances for positive polarity and 1098 instances for negative polarity. While the *Laptop* dataset has 1,335 instances for positive polarity and 998 instances for negative polarity. For the IMDB and SemEval datasets, the training sets and testing sets have already been divided. For the last two datasets, we first randomly shuffle the whole datasets and then set the training set, dev set, and test set according to 8:1:1. Our aim is to predict the sentiment polarity of a review.

Evaluation Metric. To evaluate the performance of our framework for sentiment prediction, we adopt the *Accuracy* metric, which is defined as:

$$Accuracy = \frac{\sum_{i=1}^n 1 \{y_i = \hat{y}_i\}}{N} \quad (11)$$

where y_i is the true value that the instance is labeled, while \hat{y}_i is the result predicted by our framework. N refers the total number of samples. *Accuracy* measures the percentage of correct predicted samples, from which we can evaluate our framework’s performance towards generalized dataset.

To test the performance of our framework, we employ several baselines in the experiment as indicated in [12]. For IMDB dataset, LSA, LDA, MAAS Semantic, MAAS Full, word embedding based CNN, RNN and LSTM are used. For Customer Review and Movie Review, Bag-of-Words, Vote by lexicon, Rule-based reversal, Tree-Based CRF, word embedding based RNN, CNN and LSTM are employed. While for SemEval dataset, word embedding based RNN, LSTM, GRU, Bi-GRU, and our framework without attention are used for baselines.

Model Training. In our experiments, all word vectors are initialized by GloVe⁶. The word embedding vectors are pre-trained on an unlabeled corpus with 1.2 million vocabulary size. The dimensions of word embeddings are set to 200. The weights parameters are initialized by sampling from a uniform distribution $U(-\sqrt{\frac{6}{fan-in}}, \sqrt{\frac{6}{fan-in}})$, where $fan-in$ is the units of weight tensor.

² <http://ai.stanford.edu/~amaas/data/sentiment/>.

³ <http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html>.

⁴ <http://www.cs.cornell.edu/people/pabo/movie-review-data/>.

⁵ <http://alt.qcri.org/semEval2014/task4/>.

⁶ <http://nlp.stanford.edu/projects/glove/>.

The Cross Entropy with L_2 regularization was used in this experiment as the loss function, which is defined as:

$$J(w, b) = -\frac{1}{m} \sum_{i=1}^m [y_i \log \hat{y}_i + (1 - y_i) \log (1 - \hat{y}_i)] + \frac{\lambda}{2m} \sum_{l=1}^m \|\omega\|_F^2 \quad (12)$$

where y_i denotes the ground truth; \hat{y}_i represents the predicted probability for each class. By compressing L_2 Frobenius norm, we set $\omega = 0.001$, which is the coefficient for L_2 . We trained all models with a batch size of 64 samples and the dropout rate is set to 0.2. The training results shows that using L_2 regularization and dropout strategy can better avoid overfitting.

Furthermore, we employed Adam optimization, which is provided with the advantages of RMSProp and momentum. The initial learning rate is set to 0.005, while $\beta_1, \beta_2, decay$ are set to 0.9, 0.999, 0.01 respectively.

4.2 Results and Discussion

Tables 1, 2 and 3 shows the comparison results of the proposed framework against different baselines on the four datasets.

Table 1. Accuracy for movie review and customer review

Method	Movie review	Customer review
Bag-of-word	0.764	0.814
Voting by lexicon	0.631	0.742
Rule-based reversal	0.629	0.743
Tree-CRF	0.773	0.814
Word embedding based CNN	0.778	0.819
Word embedding based RNN	0.781	0.821
Word Embedding based LSTM	0.774	0.764
Our framework	0.798	0.847

Through the comparison results, we can draw from that our framework has more superiorities compared with the baseline methods. The reason is mainly that: (1) The sentential-sequence modeling by using pre-attention Bi-GRU is better capable of coping the complicated interaction between words from either side of a sentence and capturing underlying relationships between words for learning long distance syntax dependencies. (2) The attention mechanism helps to pay close attention to the important parts of a sentence and utilize the internal keywords for improvement on classifying sentiment polarity from word-feature seizing level. (3) The particular structure of the framework not only help the behavior of sentiment analysis but also solve the vanishing and exploding problem properly.

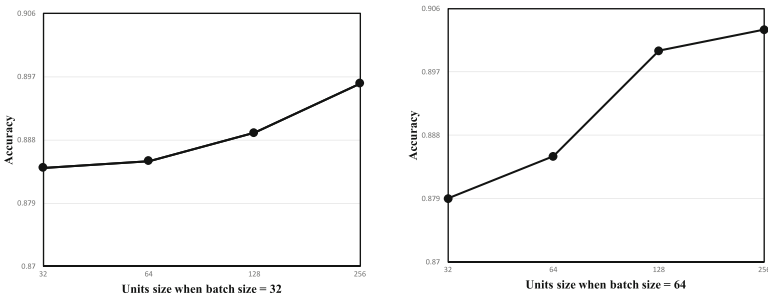
Table 2. Accuracy for IMDB

Method	IMDB
LSA	0.839
LDA	0.674
MAAS semantic	0.873
MAAS full	0.874
Word embedding based CNN	0.884
Word embedding based RNN	0.829
Word embedding based LSTM	0.835
Our framework	0.903

Table 3. Accuracy for SemEval laptop and restaurant

Method	SemEval laptop	SemEval restaurant
Word embedding based RNN	0.656	0.799
Word embedding based LSTM	0.761	0.862
Word embedding based GRU	0.783	0.844
Word embedding based Bi-GRU	0.786	0.849
Our framework (no attention)	0.793	0.864
Our framework	0.800	0.868

To achieve better performance of our framework, we have tried various hyper-parameters. We present our tuning process of batch size and units size of IMDB dataset in Fig. 3. As we can see in the figures, the accuracy increases when units size get larger, which denotes a relatively larger network structure can capture more information from the sentences and reduce bias. However, if blindly enlarging the network layers, the optimize function will be more likely fall into the local optimal solution. So we stop increase the size of the units and find that 256 is

**Fig. 3.** Accuracy for different batch and units size

the optimal choice for our framework. By comparing the two figures we can find that a bigger batch size also helps to achieve better performance. IMDB is a larger dataset, by employing mini-batch size learning and increasing batch size properly, the memory utilization improves and the training loss decrease.

We also visualize the word embedding learned from our framework in typical IMDB dataset and presents in Fig. 4. The t-SNE method [14] was employed for dimension reduction. The point in the figure represents a word embedding. From the embedding scatter figure, it is observed that the word embeddings are roughly separated into two parts and the boundary is apparent. The positive and negative instances rate is 0.5/0.5 in IMDB, and we can find the upper and lower part has similar size. The upper part mainly includes the negative words such as “terrible”, “disgusting” and “worst”. The upper part contains positive words like “wonderful”, “happy” and “excellent”. From the figure, we can conclude that words with same sentiment polarities are clustered, which indicates our framework is capable of using word embeddings to classify the sentiment polarity better.

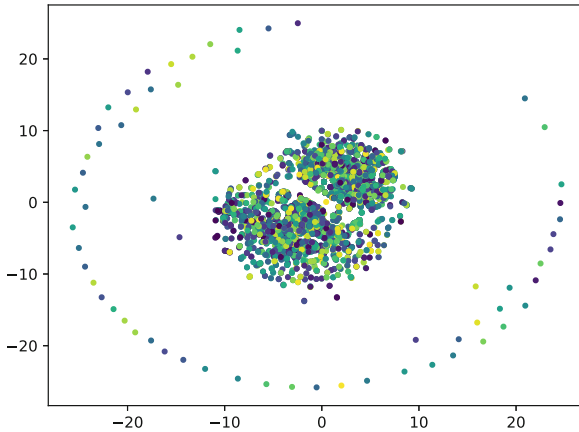


Fig. 4. Embedding learned by the proposed framework.

5 Conclusion and Future Work

In this paper, we first introduced the task of sentiment analysis and ANNs applied in this field. In detail, we introduced the advantage of RNN for learning the underlying relationships between words. However, with the limitation of the vanishing and exploding problem, LSTM and GRU begin to be used. Then, we analyzed the advantage of GRU and discussed the importance of sentential-level learning and syntax dependencies. We also highlighted the significance of focusing on the key information of an input sequence from the word-feature level. Afterward, we proposed an attention aware Bi-GRU framework. In this

work, we analyzed sentiment polarity from the aspects of sentential-sequence modeling and word-feature seizing to classify the sentiment polarity. Besides, a post-attention GRU was added for imitating the function of the decoder to extract the predicted features conditioned on the pre-attention Bi-GRU and the attention layer. The experiential study on the four standard corpora receives promising results.

Concerning the future work, we plan to use other methods to test our framework. For example, we could apply the Bi-LSTM for the sentential-modeling part. Furthermore, we project to design more powerful and flexible attention structure in the word-feature seizing level.

Acknowledgments. This work was partially supported by the National Natural Science Foundation of China (No. 61332018), and the Fundamental Research Funds for the Central Universities.

References

1. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. CoRR abs/1409.0473 (2014)
2. Bengio, Y., Simard, P.Y., Frasconi, P.: Learning long-term dependencies with gradient descent is difficult. *IEEE Trans. Neural Netw.* **5**(2), 157–166 (1994)
3. Cho, K., van Merriënboer, B., Bahdanau, D., Bengio, Y.: On the properties of neural machine translation: Encoder-decoder approaches. In: *Proceedings of 8th Workshop on Syntax, Semantics and Structure in Statistical Translation*, pp. 103–111 (2014)
4. Chung, J., Gülçehre, Ç., Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling. CoRR abs/1412.3555 (2014)
5. Ding, X., Liu, B., Yu, P.S.: A holistic lexicon-based approach to opinion mining. In: *Proceedings of 2008 International Conference on Web Search and Web Data Mining*, pp. 231–240 (2008)
6. Goodfellow, I., Bengio, Y., Courville, A., Bengio, Y.: *Deep Learning*, vol. 1. MIT press, Cambridge (2016)
7. He, L., Lee, K., Lewis, M., Zettlemoyer, L.: Deep semantic role labeling: what works and what’s next. In: *Proceedings of 55th Annual Meeting of the Association for Computational Linguistics*, vol. 1, pp. 473–483 (2017)
8. Hermann, K.M., et al.: Teaching machines to read and comprehend. In: *Proceedings of 2015 Annual Conference on Neural Information Processing Systems*, pp. 1693–1701 (2015)
9. Hochreiter, S., Schmidhuber, J.: Long short-term memory. *Neural Comput.* **9**(8), 1735–1780 (1997)
10. Kudo, T., Matsumoto, Y.: Chunking with support vector machines. In: *Proceedings of 2nd Meeting of the North American Chapter of the Association for Computational Linguistics* (2001)
11. Lafferty, J.D., McCallum, A., Pereira, F.C.N.: Conditional random fields: probabilistic models for segmenting and labeling sequence data. In: *Proceedings of 18th International Conference on Machine Learning*, pp. 282–289 (2001)

12. Liu, J., Rong, W., Tian, C., Gao, M., Xiong, Z.: Attention aware semi-supervised framework for sentiment analysis. In: Lintas, A., Rovetta, S., Verschure, P.F.M.J., Villa, A.E.P. (eds.) ICANN 2017. LNCS, vol. 10614, pp. 208–215. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-68612-7_24
13. Lu, Y., Salem, F.M.: Simplified gating in long short-term memory (LSTM) recurrent neural networks. CoRR abs/1701.03441 (2017)
14. van der Maaten, L., Hinton, G.: Visualizing data using t-SNE. *J. Mach. Learn. Res.* **9**, 2579–2605 (2008)
15. Mnih, V., Heess, N., Graves, A., Kavukcuoglu, K.: Recurrent models of visual attention. In: Proceedings of 2014 Annual Conference on Neural Information Processing Systems, pp. 2204–2212 (2014)
16. Mousa, A.E., Schuller, B.W.: Contextual bidirectional long short-term memory recurrent neural network language models: a generative approach to sentiment analysis. In: Proceedings of 15th Conference of the European Chapter of the Association for Computational Linguistics, pp. 1023–1032 (2017)
17. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up? Sentiment classification using machine learning techniques. In: Proceedings of 2002 Conference on Empirical Methods in Natural Language Processing (2002)
18. Pascanu, R., Mikolov, T., Bengio, Y.: On the difficulty of training recurrent neural networks. In: Proceedings of 30th International Conference on Machine Learning, pp. 1310–1318 (2013)
19. Qiu, L., Lei, Q., Zhang, Z.: Advanced sentiment classification of tibetan microblogs on smart campuses based on multi-feature fusion. *IEEE Access* **6**, 17896–17904 (2018)
20. Raza, K., Alam, M.: Recurrent neural network based hybrid model for reconstructing gene regulatory network. *Comput. Biol. Chem.* **64**, 322–334 (2016)
21. Schuller, B.W., Mousa, A.E., Vryniotis, V.: Sentiment analysis and opinion mining: on optimal parameters and performances. *Wiley Interdisc. Rev. Data Min. Knowl. Disc.* **5**(5), 255–263 (2015)
22. Wang, Y., Huang, M., Zhu, X., Zhao, L.: Attention-based LSTM for aspect-level sentiment classification. In: Proceedings of 2016 Conference on Empirical Methods in Natural Language Processing, pp. 606–615 (2016)
23. Wang, Y., Pal, A.: Detecting emotions in social media: a constrained optimization approach. In: Proceedings of 24th International Joint Conference on Artificial Intelligence, pp. 996–1002 (2015)
24. Zhang, B., Xiong, D., Su, J.: A GRU-gated attention model for neural machine translation. CoRR abs/1704.08430 (2017)
25. Zhang, M., Zhou, Z.: A review on multi-label learning algorithms. *IEEE Trans. Knowl. Data Eng.* **26**(8), 1819–1837 (2014)
26. Zhao, J., Liu, K., Xu, L.: Sentiment analysis: mining opinions, sentiments, and emotions. *Comput. Linguist.* **42**(3), 595–598 (2016)
27. Zhou, D., Zhang, X., Zhou, Y., Zhao, Q., Geng, X.: Emotion distribution learning from texts. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pp. 638–647 (2016)
28. Zhou, J., Xu, W.: End-to-end learning of semantic role labeling using recurrent neural networks. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, pp. 1127–1137 (2015)



Neural Sentiment Classification with Social Feedback Signals

Tao Wang¹, Yuanxin Ouyang[✉], Wenge Rong, and Zhang Xiong

School of Computer Science and Engineering,
Beihang University, Beijing 100191, China
{12061224, oyyx, w.rong, xiongz}@buaa.edu.cn

Abstract. Neural network methods have achieved promising results for document-level sentiment classification. Since the popularity of Web 2.0, a growing number of websites provide users with voting and feedback systems (or called social feedback system). However, most existing sentiment classification models only focus on text information while ignoring the social feedback signals from fellow users, despite the association between voting and review predicting. To address this issue, first, we conduct empirical analysis based on a large-scale review dataset to verify the relevance between the social feedback signals and the review predicting. Afterward, we build a hierarchical attention model to generate sentence-level and document-level representations. Finally, we feed the social feedback information into word level and sentence level attention layers. Extensive experiments demonstrate that our model can significantly outperform several strong baseline methods and social feedback signals can promote the performance of attention model.

Keywords: Sentiment classification · Social feedback signal
Attention mechanism · Recurrent neural network

1 Introduction

Document-level sentiment classification aims to analyze people's attitudes or opinions about certain topics or the overall polarity to a document. With the rapid growth of online review platforms such as Yelp, people increasingly share their experiences and views with products, services and business in the form of reviews and ratings. Researches show that consumers tend to consider the information on the online review sites as impartial message sources rather than that on commercial sites [1].

Consumers can be attracted to visit an online review site if user-generated reviews about products or business are useful. Thus, in order to promote engagement between the minority of review writers and the majority of review readers as well as encourage review creators to write helpful reviews, online communities provide users with voting and feedback systems. These systems enable fellow users to interact with review writers by voting reviews. On Yelp review site, each

review can be voted as *useful*, *funny* and *cool* (in any combination), these signals functioning to express readers' attitude towards reviews.

Most existing sentiment classification approaches especially neural network methods only take document text information as input while ignoring the relationship between review polarity and social feedback signals from review readers. We show an example of parts of a review from Recsys 2013 dataset¹ in Fig. 1. A review writer writes a review with a lot of positive words but gives one star. In this case, feedback from review readers could be helpful to review rating prediction. Therefore, it is possible to take social feedback signals such as votes of review readers into consideration to improve the performance of sentiment classification.

*Here 's the 1. 2. 3... 1. **great** food. i **love** hot new mexico style food. **good** job. 2. they have a little brewery and make 3 house beers. **unfortunately** , they 're all **shitty** . 3. they could really improve the service. 4 17 11 update some asshole who was clearly affiliated with XXX contacted me via private message to slam me for my review. i hope more people read my review and stay away from this restaurant. **good** food folks, just get it to go !*

Fig. 1. A one-star review received 4 *funny*, 1 *cool* and 4 *useful* votes. The words highlighted in red denote positive connotations, and those in blue are negative. The funny votes are related to higher likelihood of low rates though there are many positive words in this review, which probably indicates that the review readers find the lower rated reviews and those with a negative tone more humorous [2]. (Color figure online)

In this paper, first, we take a close look at the relationship between review reader's feedback and review rating on two review datasets. Then, to incorporate social feedback signals into sentiment analysis, we propose a novel hierarchical vote-aware attention neural network model. Our model mainly consists of two parts: (1) we build a hierarchical neural network based model to generate sentence-level representation and document-level representation jointly. (2) we introduce vote-aware attention to extract social feedback signals as attentions over different semantic levels of a review text.

2 Related Work

2.1 Social Feedback Signals from Fellow Users

To encourage not only the creators of helpful content but also the readers to participate in the site, a crowd of online review websites provide users with a feedback system [3], for example, Twitter's favorites or retweets, Amazon's helpful votes and Yelp's *funny*, *cool* and *useful* votes. Archak et al. [4] suggest that reviewer, social context and perceived attributes of the review text may

¹ <https://www.kaggle.com/c/yelp-recsys-2013>.

all shape consumer response to reviews. Bakhshi et al. [5] suggest that even exogenous factors such as weather and demographics of users might impact the ratings and reviews.

There is a body of work on analyzing product reviews. Lu et al. [6] use a latent topic method to extract rated quality aspects (corresponding to concepts such as “price” or “shipping”) from comments on eBay. Danescu et al. [7] suggest that the helpfulness scores on Amazon site are dependent on both the content of the review and social feedback signals from other reviews. Bakhshi et al. [3] conduct a research on Yelp review dataset to figure out whether the social signals of a review are indicative of review rating and sentiment. Although there are some studies on the relationship between votes and review rating, no previous study has looked into adding voting information to sentiment classification.

2.2 Neural Network Model for Sentiment Classification

With the trends of deep learning in natural language processing, neural network models are introduced into document categories field, especially in document-level sentiment analysis thanks to its ability of high-level feature representation [8]. Document-level sentiment classification usually aims to review rating prediction or review polarity prediction.

Review rating prediction goes beyond review polarity prediction (positive or negative) and target at predicting the numeric rating (e.g. 1–5 stars) of a given review. Many works on document classification are based on word vectors to generate the document-level representation. Kim et al. [9] propose a convolutional neural network (CNN) for document classification. Zhang et al. [10] adopt a character-level CNN for text categories. There are also some works about hierarchical neural network structure dealing with document-level sentiment classification. Tang et al. [11] put forward hierarchical recurrent neural network (RNN), outperforming shallow CNN-based models. Tang et al. [12] propose to add user and product information into neural network models. Recently, Yang et al. [13] introduce a hierarchical attention network (HAN) with gated recurrent units which outperforms traditional and neural baselines.

Despite the apparent success of neural network methods, most existing sentiment classification models only focus on text information but ignore the influences of relative pieces of information. To address this issue, we propose an novel neural network model with social feedback information to serve as attention at both word level and sentence level.

3 Methodology

3.1 Consistency Assumption Verification

RecSys 2013 Challenge Dataset contains 229,907 reviews, three kind of social feedback signals (*funny*, *cool* and *useful*) and five classes.

Table 1. p -value of Kolmogorov–Smirnov test

Dataset	<i>Funny & Not-funny</i>	<i>Cool & Not-cool</i>	<i>Useful & Not-useful</i>
RecSys 2013	2.968e−107	6.101e−48	1.148e−45

To understand the relationship between each social feedback signal and review’s rating, we use the *useful*, *funny* and *cool* votes counts as measure of Recsys 2013 review’s social feedback. We define a binary variable for each of *useful*, *funny* and *cool* votes signal instead of numerical votes values and perform three groups comparison between *useful* and *not-useful*, *funny* and *not-funny*, *cool* and *not-cool* reviews. We consider a review to be *funny/cool/useful* if it has at least 3 *funny/3 cool/3 useful* votes². We use a two-sample Kolmogorov–Smirnov test (K–S test) with the null hypothesis that both groups of *funny/cool/useful* reviews rating and *not-funny/not-cool/not-useful* reviews rating share the same distribution, results are shown in Table 1. The K-S test rejects the null hypothesis in all three groups distribution comparison, indicating that *cool* reviews rating and *not-cool* reviews rating are not coming from the same distribution, because the p -value is smaller than $2.2e-16$ which suggesting that reviews rating from *funny/cool/useful* reviews rating and *not-funny/not-cool/not-useful* reviews rating are significantly different.

3.2 Hierarchical Vote-Aware Attention Neural Network

In this section, we first introduce the Hierarchical Neural Network (HNN) model in details, discussing how to obtain sentence-level and document-level semantic representation via the HNN model. Then, we present our vote-aware attention mechanism which incorporates the social feedback signals from fellow reviewers to enhance document representation used as input of review rating prediction.

From fellow user social feedback signals of a review, we can learn additional information of review’s sentiment. We bring Hierarchical Vote-aware Attention Neural Network (HVANN) to capture social feedback signals into sentiment classification model. The overall architecture of HVANN is shown in Fig. 2. Suppose a review document d has three social feedback information *funny*, *cool* and *useful*. First we represent d with n sentence $\{\mathbf{S}_1, \mathbf{S}_2, \dots, \mathbf{S}_i, \dots, \mathbf{S}_n\}$, and define L_i to be the length of \mathbf{S}_i , then \mathbf{S}_i can be represented with $\{\mathbf{w}_1^i, \mathbf{w}_2^i, \dots, \mathbf{w}_j^i, \dots, \mathbf{w}_{L_i}^i\}$. We add *funny*, *cool* and *useful* in both word-level and sentence-level attention layers.

Hierarchical Neural Network Base Model. We adopt a hierarchical structure composed of sentence-level and document-level to obtain a semantic representation of a document. We employ bidirectional Batch-Normalized Long

² We choose three as the threshold to control for noise and weak social feedback as a result of comparative experiments (see Sect. 4.3).

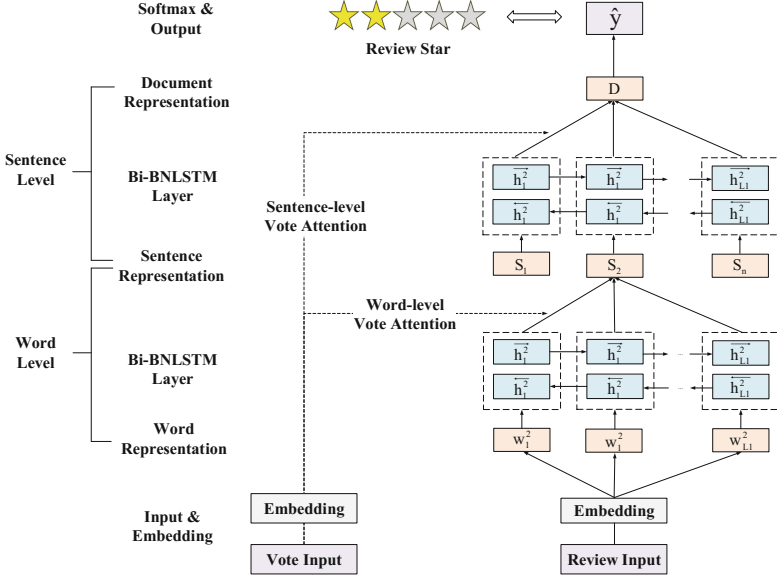


Fig. 2. Structure of hierarchical vote-aware attention neural network

Short-Term Memory (bi-BNLSTM) as recurrent neurons on both word-level and sentence-level layers. Our bi-BNLSTM inspired by Laurent’s recurrent batch normalization unit [14]. Batch normalization (BN) aims to reduce internal covariate shift [15]. The batch normalizing transform used in bi-BNLSTM unit is as follows:

$$BN(\mathbf{I}; \gamma) = \gamma \circ \frac{\mathbf{I} - \hat{E}(\mathbf{I})}{\sqrt{\hat{Var}(\mathbf{I}) + \epsilon}} \quad (1)$$

where \mathbf{I} is the input, γ is parameter that determine the mean deviation of the normalized input, ϵ is a regularization hyperparameter. During training, the statistics $\hat{E}(\mathbf{I})$ and $\hat{Var}(\mathbf{I})$ are estimated by the sample mean and sample variance of the current mini-batch. The bi-BNLSTM unit is implemented as the following:

$$\begin{bmatrix} \hat{\mathbf{i}}_j^i \\ \hat{\mathbf{f}}_j^i \\ \hat{\mathbf{o}}_j^i \end{bmatrix} = \text{sigmoid}(BN(\mathbf{w}_j^i \mathbf{U}_x; \gamma_x) + BN(\mathbf{h}_{j-1}^i \mathbf{W}_h; \gamma_h) + \mathbf{b}_h) \quad (2)$$

$$\hat{\mathbf{c}}_j^i = \tanh(BN(\mathbf{w}_j^i \mathbf{U}_x; \gamma_x) + BN(\mathbf{h}_{j-1}^i \mathbf{W}_h; \gamma_h) + \mathbf{b}_c) \quad (3)$$

$$\mathbf{c}_j^i = \mathbf{c}_{j-1}^i \circ \hat{\mathbf{f}}_j^i + \hat{\mathbf{c}}_j^i \circ \hat{\mathbf{i}}_j^i \quad (4)$$

$$\mathbf{h}_j^i = \tanh(BN(\mathbf{c}_j^i; \gamma_c)) \circ \hat{\mathbf{o}}_j^i \quad (5)$$

In BNLSTM cell, given an input word embedding \mathbf{w}_j^i , at word level, at each step, the hidden state \mathbf{h}_j^i and the current cell state \mathbf{c}_j^i will be updated with the

previous hidden state \mathbf{h}_{j-1}^i and the previous cell state \mathbf{c}_{j-1}^i as formulas (2)–(5), where \mathbf{i}_j^i , \mathbf{f}_j^i and \mathbf{o}_j^i are gate activations, \circ is element-wise multiply operation, those \mathbf{b} , \mathbf{U} and \mathbf{W} are trainable parameters. \mathbf{w}_j^i is the j -th input word vector of i -th sentence, \mathbf{h}_j^i is the $(j-1)$ -th output of i -th sentence.

In order to increase input information of review text to the network, we adopt bidirectional method to model review semantics both from forward and backward. The bi-BNLSTM is defined as a concatenation of the outputs of two BNLSTM cells, the forward cell and the backward cell. For sequence input vectors $\{\mathbf{w}_1^i, \mathbf{w}_2^i, \dots, \mathbf{w}_j^i, \dots, \mathbf{w}_{L_1}^i\}$, the forward cell reads sequence from \mathbf{w}_1^i to $\mathbf{w}_{L_1}^i$ while the backward cell from $\mathbf{w}_{L_1}^i$ to \mathbf{w}_1^i . We concatenate the forward hidden state $\overrightarrow{\mathbf{h}}_j^i$ and the backward hidden state $\overleftarrow{\mathbf{h}}_j^i$, then get the bi-BNLSTM output $\mathbf{h}_j^i = [\overrightarrow{\mathbf{h}}_j^i, \overleftarrow{\mathbf{h}}_j^i]$ corresponding to the input \mathbf{w}_j^i . We finally feed hidden states $\{\mathbf{h}_1^i, \mathbf{h}_2^i, \dots, \mathbf{h}_{L_1}^i\}$ to the word-level vote-aware attention layer to obtain the sentence representation \mathbf{S}_i .

At document level, we also feed the sentence representations $\{\mathbf{S}_1, \mathbf{S}_2, \dots, \mathbf{S}_n\}$ into bi-BNLSTM cells and generate the document-level representation d through the sentence-level vote-aware attention layer in a similar way.

Vote-Aware Attention. We have demonstrated that review ratings with different votes (e.g. *funny* or *not-funny*) have different distributions. We assume that not all words contribute equally to the sentence meaning for different social feedback distribution. Hence, we bring in vote-aware attention mechanism which incorporates the social feedback signals from fellow reviewers to enhance document representations. Specifically, we adopt word-level vote-aware attention to generate sentence representations and sentence-level vote-aware attention to do with the document representation. Detailed implementations are as follows:

$$\mathbf{u}_j^i = \tanh(\mathbf{h}_j^i \mathbf{W}_h + \mathbf{v} \mathbf{U}_\gamma + \mathbf{b}) \quad (6)$$

$$\alpha_j^i = \frac{\exp(\mathbf{M} \mathbf{u}_j^i)}{\sum_{j=1}^{L_i} \exp(\mathbf{M} \mathbf{u}_j^i)} \quad (7)$$

$$\mathbf{S}_i = \sum_{j=1}^{L_i} \alpha_j^i \mathbf{h}_j^i \quad (8)$$

where \mathbf{M} , \mathbf{W}_h and \mathbf{U}_γ are weight matrices we need to train. We embed each kind of social feedback signal \mathbf{v} (we consider *funny*, *useful* but *not-cool* to be one kind of social feedback signal, *funny*, *useful* and *cool* to be another, that is to say, we divide reviews into 8 kinds) as continuous and real-valued vectors $\mathbf{v} \in \mathbb{R}^{d_v}$ where d_v is the dimensions of vote embeddings. First, we feed the bi-BNLSTM hidden state \mathbf{h}_j^i and social feedback vector \mathbf{v} through a fully connected layer to get \mathbf{u}_j^i as a hidden representation of the j -th word in i -th sentence of review document. Then we get α_j^i as attention weight of each word. Finally, we can get the enhanced sentence representation \mathbf{S}_i by a weighted sum of hidden states.

In sentence level, we also use vote-aware attention mechanism to aggregate the representation of sentences to form the document representation \mathbf{d} as follows:

$$\mathbf{d} = \sum_{i=1}^n \beta^i \mathbf{h}^i \quad (9)$$

where β^i is the attention weight of each sentence and \mathbf{h}^i is the word-level bi-BNLSTM hidden state. β^i can be calculated similarly to the word-level attention. The document representation with social feedback signals could perform better in extracting semantic information.

Sentiment Classification. The document-level representation \mathbf{d} is of high level of review semantics. Hence we regard it as features for review rating classification. We use a linear layer with softmax function to project document-level representation \mathbf{d} into the target space of Y star classes:

$$\hat{p}_y = \text{softmax}(\mathbf{W}_y \mathbf{d} + \mathbf{b}_y) \quad (10)$$

where \mathbf{W}_y and \mathbf{b}_y are trainable weight and bias. \hat{p}_y is the predicted probability of review star class y . We use cross-entropy error between review star labels and our model’s predict star labels as training loss:

$$\text{Loss} = - \sum_{y=1}^Y p_y \log(\hat{p}_y) \quad (11)$$

where p_y is the review star probability of review star class y with ground truth being 1 and others being 0.

4 Experiments

4.1 Datasets and Evaluation Metrics

In order to evaluate the model we proposed and test the performance of vote-aware attention, we conduct experiments on two sentiment classification datasets with voting information. Existing benchmark datasets for sentiment classification typically do not contain social feedback signals, thus, we acquire our data from RecSys 2013 Challenge and Yelp 2016 Academic Challenge³. The statistics of the balanced datasets are summarized in Table 2, *funny*/review means average *funny* votes each review received. In this research, we randomly split the datasets into ten sets and adopt the 10-fold cross-validation strategy to compute the average accuracy. We use standard *Accuracy* defined as follows to measure the overall sentiment classification performance.

$$\text{Accuracy} = \frac{\sum_{i=1}^n 1\{y_i = p_i\}}{N} \quad (12)$$

where N is the number of test review documents, y_i stands for the true value that the review is labeled and p_i is the result predicted by the neural network.

³ https://www.yelp.com/dataset_challenge.

Table 2. Statistics of RecSys 2013 and Yelp 2016

Dataset	Classes	Reviews	<i>Funny</i> /review	<i>Cool</i> /review	<i>Useful</i> /review
RecSys 2013	5	229,907	0.699	0.868	1.387
Yelp 2016	5	2,685,066	0.431	0.539	1.007

4.2 Experimental Settings

We use the 300-dimensional pre-trained Google News word embeddings⁴, the word embeddings are fine-tuned on the training set. Adadelta [16] is adopted for parameters updating. We select the best parameters configuration based on performance on the validation set. We benchmark several strong baseline methods for document-level sentiment classification to test the performance of the proposed model. Paragraph-Vector [17], FastText [18], CNN-word [9], CNN-char [10], word-based bi-LSTM and word-based Hierarchical bi-LSTM serve as baselines. Pre-trained word embeddings also be used in word-based models the same with our model.

4.3 Results and Discussions

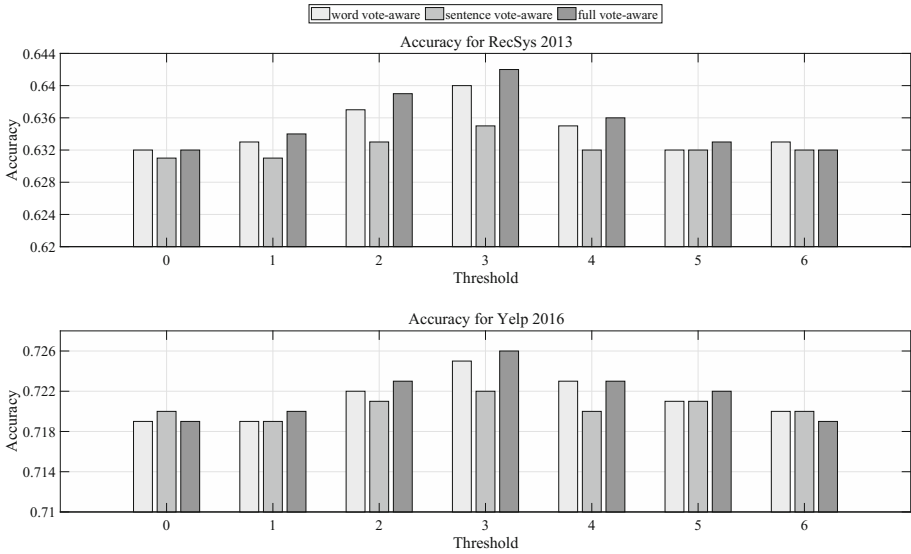
Result Analysis. Experimental results are listed in Table 3, where RecSys 2013 and Yelp 2016 are 5 class classification tasks and RecSys 2013 P. and Yelp 2016 P. are binary classification tasks predicting polarity labels by considering review star classes 1 and 2 negative, and 3, 4 and 5 positive. It is found that our models outperforms the baseline methods in all datasets. The reasons for such phenomena mainly contain four main folds: (1) the Recurrent neural network model performances well in learning long-range dependencies of words because of the ability to catch long sequence information. (2) the attention mechanism makes the hierarchical model learn more meaningful words and sentences. (3) Comparing models with and without bi-BNLSTM unit, we can see that reducing internal covariate shift show significant improvements. (4) the social feedback information helps attention layer to get more informative sentence/document representations combined with review context and votes information.

Besides, our model with social feedback signals performs better in all datasets than models without social feedback information. The reason is that other baseline methods cannot learn the underlying relationships between social feedback signals and review rating. Additionally, comparing models with word vote-aware, sentence vote-aware and full vote-aware, we can see that both word and sentence level vote-aware can promote classification performance. This confirms that social signals used in our framework can be helpful to capture the context-dependent word/sentence importance. The reason is that different distributions of review (validated by K-S test in Sect. 3.1) grouped by social feedback signals contain internal information useful to contribute a more informative sentence/document representation.

⁴ <https://code.google.com/archive/p/word2vec/>.

Table 3. Accuracy for RecSys 2013 and Yelp 2016

Dataset	RecSys 2013	RecSys 2013 P.	Yelp 2016	Yelp 2016 P.
Paragraph-Vector	0.583	0.885	0.676	0.919
FastText	0.590	0.907	0.687	0.923
CNN-char	0.625	0.927	0.716	0.942
CNN-word	0.617	0.919	0.704	0.935
bi-LSTM	0.613	0.918	0.698	0.933
Hierarchical bi-LSTM	0.622	0.926	0.709	0.942
Our (no vote-aware)	0.633	0.932	0.720	0.948
Our (word vote-aware)	0.640	0.935	0.725	0.950
Our (sentence vote-aware)	0.635	0.932	0.722	0.949
Our (full vote-aware)	0.642	0.936	0.726	0.950

**Fig. 3.** Comparisons of different threshold used for classification

Furthermore, we compared different thresholds used for control noise and weak social feedback signals. The result is shown in Fig. 3, with the threshold of social feedback signals varying from 0 to 6 (most samples get less than six votes of each kind, so the threshold range from zero to six), the classification accuracy of two databases climax at the threshold of three, the median of the range. The reasons for such phenomena probably are as follows. (1) If the threshold is set to be too small, the evaluation of social feedback signals might be affected by weak social feedback signals, rendering the model fail to obtain valid information from reviews of certain styles. (2) If the threshold is set to be too large, on the other hand, the decrease of style-specific comments could make the network

more easily influenced by noise data, leading to less improvement of sentiment classification task.

Case Study. To demonstrate the capability of our vote-aware attention, we provide a review in RecSys 2013 datasets as an example (Considering the length of the thesis, some less significant words and sentences has been removed to simplify the expressions). We visualize the word-level and sentence-level attention weights of two kinds of vote distribution (one is 3 *funny*, 0 *cool* and 4 *useful*, the other one is 0 *funny*, 0 *cool* and 4 *useful*) and basic attention which calculates the attention weights without considering the global information of social feedback signals in Fig. 4. Note that darker color means higher weights, and the first column represents the attention weights of each sentence.

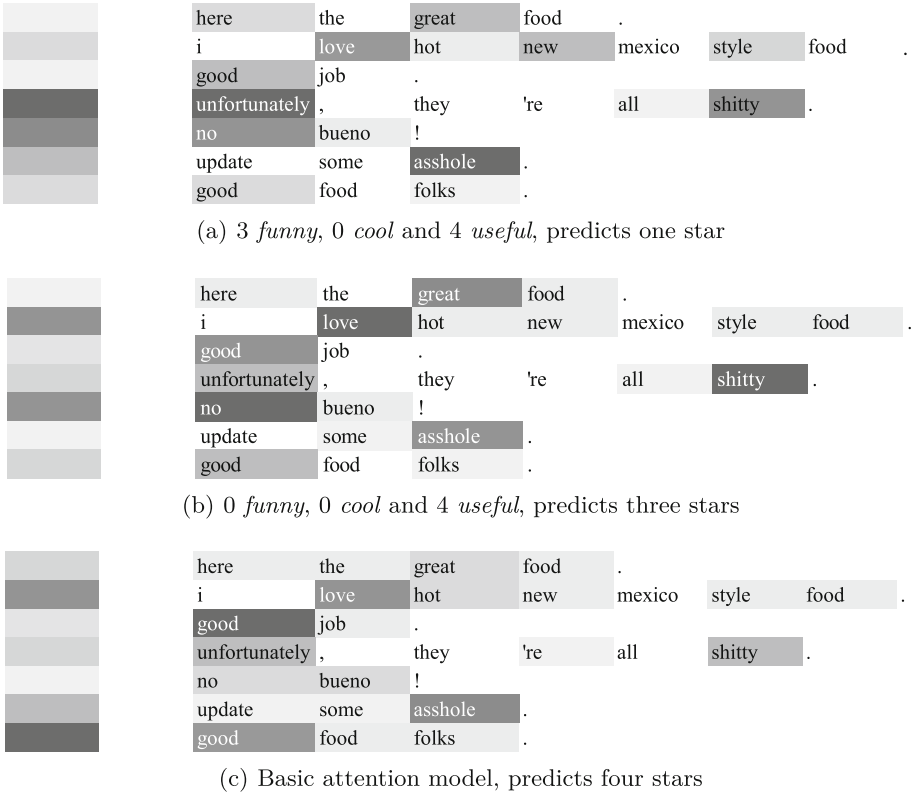


Fig. 4. Visualization of attention weights over words and sentence

As shown in Fig. 4(a), with 3 *funny*, 0 *cool* and 4 *useful* votes given, our vote-aware model focuses on the turning word “unfortunately” at the word-level attention and the turning sentence “unfortunately, they’re all shitty” at

the sentence-level attention, finally predicts the correct label of one star. This indicates our model pays more attention to positive or negative turning words when the review gets 3 *funny* votes. Figure 4(b) shows, with 0 *funny*, 0 *cool* and 4 *useful* votes given, our vote-aware model focuses on both positive and negative words, giving a three-star prediction. In Fig. 4(c), the model without vote-aware attention mainly focuses on positive words/sentences in word-level and sentence-level attention, giving a positive four-star prediction.

5 Conclusion

In this paper, we validate the underlying relationship between review rating and social feedback signals. Afterwards, we introduce a hierarchical vote-aware attention model for review rating classification under supervised learning. With the vote-aware attention, our model can take account of the global social feedback signals in both word level and sentence level. In experiments, we evaluate our model on large-scale review datasets. The experimental results show that our proposed framework achieves consistent improvements compared with other strong baseline models.

In future work, we plan to explore the relationship between review writers and social feedback signals. And we will replace recurrent neural network with other sequence model.

Acknowledgements. This work was partially supported by the National Natural Science Foundation of China (No. 61332018), and SKLSDE project under Grant No. SKLSDE-2017ZX.

References

1. Freedman, S., Jin, G.Z.: The information value of online social networks: lessons from peer-to-peer lending. *Int. J. Ind. Organ.* **51**, 185–222 (2017)
2. Bakhshi, S., Kanuparth, P., Shamma, D.A.: Understanding online reviews: funny, cool or useful? In: CSCW, pp. 1270–1276. ACM (2015)
3. Bakhshi, S., Kanuparth, P., Shamma, D.A.: If it is funny, it is mean: understanding social perceptions of yelp online reviews. In: GROUP, pp. 46–52. ACM (2014)
4. Archak, N., Ghose, A., Ipeirotis, P.G.: Show me the money!: deriving the pricing power of product features by mining consumer reviews. In: KDD, pp. 56–65. ACM (2007)
5. Bakhshi, S., Kanuparth, P., Gilbert, E.: Demographics, weather and online reviews: a study of restaurant recommendations. In: WWW, pp. 443–454. ACM (2014)
6. Lu, Y., Zhai, C., Sundaresan, N.: Rated aspect summarization of short comments. In: WWW, pp. 131–140. ACM (2009)
7. Danescu-Niculescu-Mizil, C., Kossinets, G., Kleinberg, J.M., Lee, L.: How opinions are received by online communities: a case study on amazon.com helpfulness votes. In: WWW, pp. 141–150. ACM (2009)
8. Liu, B.: *Sentiment Analysis and Opinion Mining*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers, San Rafael (2012)

9. Kim, Y.: Convolutional neural networks for sentence classification. In: EMNLP, pp. 1746–1751, ACL (2014)
10. Zhang, X., Zhao, J.J., LeCun, Y.: Character-level convolutional networks for text classification. In: NIPS, pp. 649–657 (2015)
11. Tang, D., Qin, B., Liu, T.: Document modeling with gated recurrent neural network for sentiment classification. In: EMNLP, pp. 1422–1432. The Association for Computational Linguistics (2015)
12. Tang, D., Qin, B., Liu, T.: Learning semantic representations of users and products for document level sentiment classification. In: ACL (1), pp. 1014–1023. The Association for Computer Linguistics (2015)
13. Yang, Z., Yang, D., Dyer, C., He, X., Smola, A.J., Hovy, E.H.: Hierarchical attention networks for document classification. In: HLT-NAACL, pp. 1480–1489. The Association for Computational Linguistics (2016)
14. Laurent, C., Pereyra, G., Brakel, P., Zhang, Y., Bengio, Y.: Batch normalized recurrent neural networks. In: ICASSP, pp. 2657–2661. IEEE (2016)
15. Ioffe, S., Szegedy, C.: Batch normalization: accelerating deep network training by reducing internal covariate shift. In: ICML. JMLR Workshop and Conference Proceedings, vol. 37, pp. 448–456. JMLR.org (2015)
16. Zeiler, M.D.: ADADELTA: an adaptive learning rate method. CoRR abs/1212.5701 (2012)
17. Le, Q.V., Mikolov, T.: Distributed representations of sentences and documents. In: ICML. JMLR Workshop and Conference Proceedings, vol. 32, pp. 1188–1196. JMLR.org (2014)
18. Grave, E., Mikolov, T., Joulin, A., Bojanowski, P.: Bag of tricks for efficient text classification. In: EACL (2), pp. 427–431. Association for Computational Linguistics (2017)



A Concept for Generating Business Process Models from Natural Language Description

Krzysztof Honkisz, Krzysztof Kluza^(✉), and Piotr Wiśniewski

AGH University of Science and Technology,
al. A. Mickiewicza 30, 30-059 Krakow, Poland
honkisz@student.agh.edu.pl, {kluza,wpiotr}@agh.edu.pl

Abstract. Manual extraction of business process models from technical documentation is a time-consuming task. Several approaches to generating such process models have been proposed. We present a proposal of a new method for extracting business process from natural language text through intermediate process model using the spreadsheet-based representation. Such intermediate model is transformed into a valid BPMN process model. Our method is enhanced with semantic analysis of the text, allows for quick check of the transformation result and manual correction during this process. As the obtained BPMN model is structured, it is easier to check its correctness.

1 Introduction

Business process management plays an important part in modern corporation and enterprise management. Business process models can be used as a documentation for work-flow implementation, partial automation or optimization of process. One of the most popular standards providing graphical representation of processes is Business Process Model and Notation (BPMN) [1].

Usually, some knowledge about the processes and work-flows in the company exists either in the form of human knowledge or as a textual documentation. Manual extraction of a process model from technical documentation (textual description) is a time-consuming task. Since every enterprise must constantly improve its services, their process models must be frequently updated. Moreover, manually designed models can be different, depending on the designer's experience and knowledge.

To solve this problem, several approaches of automatic business process models generation have been proposed in recent years [2–4]. An effective way of machine-aided transformation from a semi-formal document into a process model can provide a significant savings in time, additionally making maintenance of formal process models and documentation easier. Furthermore, an automatic tool

The paper is supported by the AGH UST research grant.

for model extraction can be very useful for people, who do not have the sufficient knowledge and expertise in the process modelling field.

In this paper, we present a concept of a new method for extracting business process from natural language text through intermediate process model based on the spreadsheet representation. The overview of the approach is presented in Fig. 1. The intermediate model uses our structured spreadsheet-based representation for describing business process models. The method of obtaining this model is based on the syntactic analysis of a given natural text and extracting Subject-Verb-Object construct (SVO), which can be later transformed into process activities. Our method is enhanced with semantic analysis of the text, which allows to filter out unnecessary SVO constructs and transform them into valid activities names. Then, the spreadsheet representation is transformed into a BPMN process model.

The method was implemented using our `bpmn.python` library and tested on a set of natural language business process descriptions, gathered from different BPMN tutorials and academic sources. Thanks to this intermediate step, our method allows for quick check of the transformation result and manual correction of the spreadsheet, as well as the obtained result is a structured BPMN model what can help in correctness checking and verification.

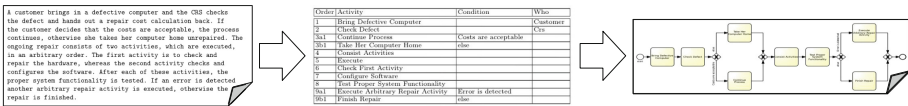


Fig. 1. Overview of the presented approach

The rest of this paper is organized as follows: Sect. 2 presents the related works in the field of process model extraction from textual description. In Sect. 3, we give an overview of our solution. Section 4 shows a case study example. The paper is summarized in Sect. 5.

2 Related Works

Terins and Thaler performed an analysis of current state-of-the-art in the field of mining process models from natural language text [3]. They presented several approaches, some of which worked with some form of structured text (use cases, group stories), some with natural language description.

Ghose, Koliadis and Chueng [5] proposed an approach for discovering a process model from text artefacts, which are described as *documents such as memos, manuals, requirements documents, design documents, mission/vision statements, meeting minutes etc.* An extraction is performed by text pattern search (for example if/then pair, which indicates a conditional flow) and POS (*Part-Of-Speech*) tagging combined with shallow parsing, which produces a syntax tree.

This approach allows to discover parts of a larger model (called *proto-models* by authors), rather than complete and sound model. Generated *proto-models* can be compared in order to find similarities and remove redundant parts.

Goncalves et al. [6] presented a technique of obtaining process models from group stories. In the first phase, a text is tokenized in order to select words which can be useful for work-flow generation. Next, a POS tagging is performed. Finally, relevant entities are identified using a set of predefined patterns. The produced BPMN model is not necessarily complete and sound – it is assumed, that it will be later improved by process designer and team members, who created the group stories.

Friedrich et al. [7] proposed an advanced approach, which uses a textual description of model. Such a description must follow some requirements – a text cannot contain any questions and the described execution of a process must be sequential (any non-sequential jumps must be explicitly made). In the first step, a syntactical analysis (called *Sentence Level Analysis* in the article) is performed, using Stanford NLP tools. Next, the semantic analysis (*Text Level Analysis*), using WordNet and FrameNet databases, allows to identify relevant entities. Finally, the process model is generated (*Process Model Generation* phase). The generated output is a sound and complete BPMN model, enriched with many additional elements (such as lanes, data sources), thanks to the rich text analysis.

Several other methodologies for transforming natural language text into formalised models were proposed. As there are various notations for representing process models [8], the related approach not always use the BPMN model. Yue, Briand and Labiche [9] presented an automated approach to transform use case descriptions to UML Activity diagrams. This methodology requires that the use case descriptions has to follow some restriction rules. These rules can be classified into two groups – the first group specifies constraints on the use of natural language, the second are requirement on the use of specific keywords to indicate the existence of control structures. In addition, the use case description explicitly lists all of the flows in the process (main and alternative) and each flow is a step-by-step description of a process.

Another approach in the field of generating formal models form natural language specification, proposed by Njonko and Abed [10], uses Semantics of Business Vocabulary and Business Rules (SBVR) as an intermediate layer for this transformation approach. It is suggested that using formalised model as an intermediate layer (in this case SBVR), it is possible to easily extend this approach for multiple models. The article presents an example of transformation from natural language business description into SQL executable query, which produces a database table that corresponds to business requirements.

As SBVR is in fact a textual description, but using controlled structured language, in our previous approach, we also developed the approach which extracts process models from the SBVR description [4]. A structured text description can be generated by searching natural language documents for keywords related to business process models. Ferreira, Thom and Fantianto [2] propose a method which is based on a syntactic analysis of natural text such as forms, e-mail messages and reports in order to generate a tagged document. This step is followed

by a logical analysis that uses a set of predefined rules to identify flow objects and swimlanes of the model which is a basis for process-oriented structures.

Identification of business process objects in natural language texts may also point out elements missing in the textual description. Therefore, preprocessing the process specification may facilitate the modeling phase. According to the survey [11], over 60% of experienced BPMN modelers find creating a business process model easier if a rule-mapped text is used as a specification, in comparison with natural language descriptions.

Structured forms generated as a result of natural language processing may suffer from various inconsistencies when using textual specifications from different sources, such as user manuals, instructions and standards. These documents may provide incomplete or contradictory information which in the extreme case may lead to the incapability of generating a correct model. To overcome this limitation, an idea of semantic unification was presented in [12]. The feature structures identified in the text can be mapped to attribute-value matrix objects which are then unified to a single description.

A comparative analysis of selected NLP-based process modeling approaches was performed by Riefer, Ternis and Thaler [3]. The authors compare the approaches based on three common pillars, namely: textual input, text analysis and model generation. The obtained results show that although most of the existing approaches do not generate complete BPMN models, they provide a firm basis for further process modeling.

Table 1 present a comparison of the mentioned approaches regarding their input and output data and the ability to generate BPMN diagrams.

Table 1. Comparison to the existing approaches

Approach	Input	Output	Method	BPMN support
Yue et al. [9]	Use case descriptions	UML activity diagram	Rule-based	○
Njoko and Abed [10]	NL specification	SQL query	SBVR	○
Ghose et al. [5]	Text documents	Proto-models	Text pattern search	◐
Ferreira et al. [2]	Natural text	Process structure	Rule-based	◑
Sokolov et al. [12]	Natural text	Unified description	SVM structures	◑
Kluza and Honkisz [4]	SBVR description	Process model	Rule-based	●
Goncalves et al. [6]	Group stories	BPMN model	POS tagging	●
Friedrich et al. [7]	Sequential description	BPMN model	Semantic analysis	●
Our approach	Natural text	BPMN model	Spreadsheet-based	●

3 Algorithm for Generating Business Process Models

This section describes the proposed approach to business process model generation from natural language description. This approach can be divided into the following steps:

1. Participants extraction – in this step, a sentence from a given description is analysed and the information about possible participants (people, systems or organizations which performs the tasks) in process are extracted,
2. Subject-verb-object constructs extraction – a sentence from given description is analysed in search of basic SVO constructs, which later will be used to create appropriate BPMN elements,
3. Gateway keywords search – a process description is analysed in search of the keywords that signalizes the presence of conditional (exclusive or inclusive) and parallel gateways,
4. Intermediate process model generation – an intermediate model in the form of spreadsheet-based representation is created from the acquired data,
5. BPMN diagram generation – a BPMN diagram is generated from the intermediate process model.

Figure 2 shows the overview of proposed approach. The generated intermediate model is parsed to BPMN diagram, using functionality provided by `bpmn_python` library. The prototypical tool implemented for the purpose of this paper generates both spreadsheet-based intermediate model and BPMN diagram, which makes the result analysis easier.

3.1 Participants Extraction

In the first step, each sentence of the description is analysed in search of words representing participants. This process is divided into three parts: (1) the sentence is analysed in search of specific dependency relations, namely nominal subject and nominal subject passive; (2) the sentence is searched for conjunction dependencies because the participant might be labelled as an object of the phrase; (3) simple semantic analysis is used to decide, whether the extracted words can be used as participants of process. The extracted word is added to output as a participant if it fulfills such conditions as the word is a pronoun or relative pronoun, one of its hypernyms belongs to the specified list of admissible hypernym keywords like `person` or `organization` or the word is a special keyword like `ATM`, `CRM`, etc.

A full name of the participant is extracted from its syntax sub-tree, provided that a given token from sub-tree is labelled with a correct dependency (see an example in Table 2 based on the syntax tree from Fig. 3).

3.2 Subject-Verb-Object Extraction

After extracting the participants from the sentence, syntactic analysis in search of SVO (subject-verb-object) constructs is performed. These construct are used to generate intermediate process model.

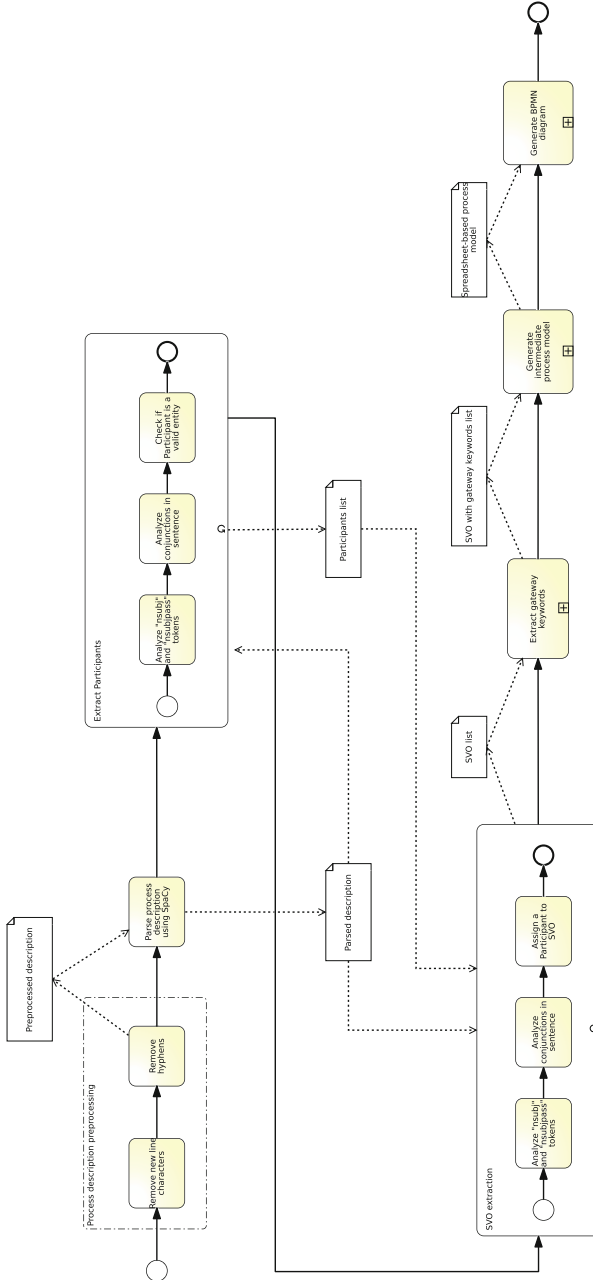


Fig. 2. BPMN diagram with overview of proposed approach.

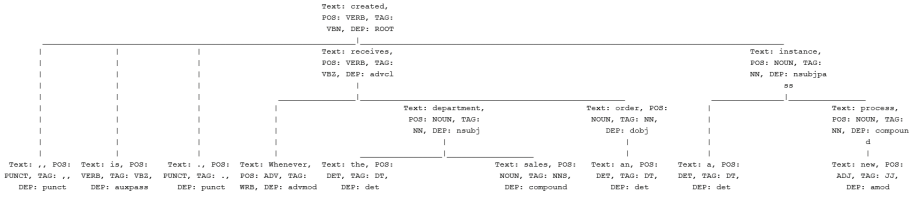


Fig. 3. A syntax tree for the simple phrase “Whenever the sales department receives an order, a new process instance is created.”

Table 2. Spreadsheet-based description generated from the sentence in Fig. 3

Order	Activity	Condition	Who
1	Receive order		Sales department
2	Create process instance		

First, the sentence is searched for nominal subject and nominal subject passive dependencies. For every word found, a new SVO construct is added to the output. In the case of words with nominal subject dependency, the subject is created from the extracted word, its predecessor in the syntax tree acts as a verb and the object is extracted from the subject’s ancestors in syntax tree.

If the appropriate token is found, it is added as an object to SVO, otherwise the object part of SVO construct is omitted. In case of tokens with nominal subject passive dependency, the object is omitted. For example, in the sentence: “Purchase is registered”, the word “Purchase” is tagged as “nsubjpass” and no object is present.

Similarly to the participants extraction, the SVO extraction also analyses the existence of conjunction in sentences. This approach helps to deal with the sentences like: “If the storehouse has successfully reserved or backordered every item”. In this case, by conjunction analysis it is possible to extract construct “backordered every item”, which is conjoined by word “backordered”.

An example of SVO extraction is shown in Table 3.

Table 3. Spreadsheet-based description generated from the sentence in Fig. 4

Order	Activity	Condition	Who
1	Ship bicycle		Sales department
2	Finish process instance		Sales department

3.3 Gateway Keywords Search

After extracting the participants and subject-verb-object constructs, the description is analysed once more, in order to find keywords indicating the existence of

possible gateway. This function searches for three different types of keywords: conditional, parallel and default flow. The first type can be later translated either into an exclusive or inclusive gateway, second – into a parallel one. Third type might be used as a default flow of conditional gateway, provided that the correspondence with a conditional keyword will be found during model generation. If no correspondence is found, the SVO will be treated as a simple activity.

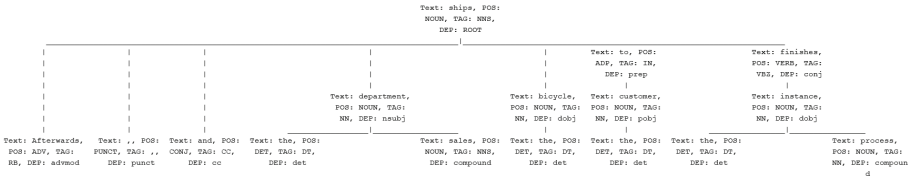


Fig. 4. Syntax tree for simple phrase “*Afterwards, the sales department ships the bicycle to the customer and finishes the process instance*”

An example of conditional flow extracted from the process description is shown in Table 4. Because the keyword associated with the default flow was found and an activity with condition *else* was added, an XOR gateway was added to the diagram (Fig. 5).

A customer brings in a defective computer and the CRS checks the defect and hands out a repair cost calculation back. If the customer decides that the costs are acceptable, the process continues, otherwise she takes her computer home unrepaired.

Text 1.1: Fragment of a process description with extractable conditional gateway

Table 4. Spreadsheet-based description generated from a text description shown in Text 1.1

Order	Activity	Condition	Who
1	Bring defective computer		Customer
2	Check defect		CrS
3a1	Continue process	Costs are acceptable	
3b1	Take her computer home	else	

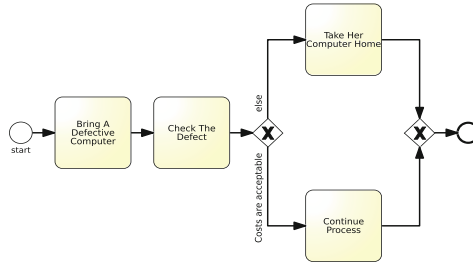


Fig. 5. BPMN diagram with an XOR gateway, generated from the intermediate process model shown in Table 4

3.4 Intermediate Model Description

The prototype implementation uses a spreadsheet-based process description [13], which employs a CSV (Comma-Separated Values) file format to represent a business process model. A business process is described by a spreadsheet table. Each row represents a single phase, which can be translated into a BPMN task or sub-process. Columns represent the properties of each phase¹, such as:

1. *Order* – the number of the corresponding phase (with the suitable suffixes for parallel or excluding tasks, and nested gateways).
2. *Activity* – the name of the performed action. There is a special case – `goto X` which signalsizes a skipped part of the process or a loop.
3. *Condition* – the condition which has to be fulfilled in order to perform the task. This property is used to implement the exclusive and inclusive gateway.
4. *Who* – the name of participant (person, system or department) responsible for executing this phase.

The spreadsheet-based process description supports only basic BPMN elements. However, the subset of supported BPMN elements covers the most commonly used elements of BPMN diagram [14].

Based on the results from the previous phases, our tool iterates over a list of extracted SVO. The process starts with a start event, and then for each construct, the appropriate action should be performed:

- If the SVO is labelled with a conditional gateway keyword, it is added to the intermediate model as a part of conditional gateway. The property *Condition* is filled with a full name of the condition SVO. If the condition SVO has a participant attached and it is not a pronoun, the full name of participant is entered as the *Who* property. Otherwise, it is left empty.
- SVO labelled with a parallel gateway keyword is treated in a similar way. However, after initialization, the next SVO from a list is added as a parallel task to the current SVO.

¹ For the sake of clarity, we focus on the four main properties of this representation.

- If the SVO is labelled as default flow, the behaviour of function depends on previously detected gateways. If the parallel gateway was found, the SVO is added as another task in this gateway. For the conditional gateway, a default flow is added – an additional task, with keyword `else` as a *Condition* column.
- If none of the previous options were executed, the SVO is added as a simple task, connected by a sequence flow.

After all of the subject-verb-objects constructs are processed, the conditional gateway flag is validated once more – similarly to the last case, if conditional gateway has only one conditional flow, a default flow is added. Finally, the end event is added, which finishes the intermediate process generation.

3.5 BPMN Diagram Generation

The spreadsheet-based model is used to generate a BPMN diagram, using functionality provided by `bpmn_python`² – our library written in Python, in order to provide a functionality to import and export BPMN diagrams in the XML format. It provides a functionality for importing spreadsheet-based model description and allows a user to export the imported diagram (stored as a graph structure) into a valid BPMN 2.0 XML file. Using these functionalities, the intermediate model is translated into a diagram, what finishes our process of translating a natural language description into a BPMN diagram.

A prototype of the proposed method was implemented in Python using SpaCy library, which provides Natural Language Processing tools (syntax parser, WordNet lexical database API). This prototype was tested against a test set of natural language business process descriptions, gathered from a few academic sources. One of such examples is presented in the following section.

4 Case Study Example

Let us consider a case study example of computer repair from BPM Academic Initiative³ presented in Text 1.2.

A customer brings in a defective computer and the CRS checks the defect and hands out a repair cost calculation back. If the customer decides that the costs are acceptable, the process continues, otherwise she takes her computer home unrepaired. The ongoing repair consists of two activities, which are executed, in an arbitrary order. The first activity is to check and repair the hardware, whereas the second activity checks and configures the software. After each of these activities, the proper system functionality is tested. If an error is detected another arbitrary repair activity is executed, otherwise the repair is finished.

Text 1.2: Text description for the computer repair case study example

² See: <https://github.com/KrzyHonk/bpmn-python>.

³ See: <https://bpmmai.org/BPMAcademicInitiative/CreateProcessModels>.

Table 5. Spreadsheet-based description for process model obtained from Text 1.2

Order	Activity	Condition	Who
1	Bring defective computer		Customer
2	Check defect		Crs
3a1	Continue process	Costs are acceptable	
3b1	Take her computer home	else	
4	Consist activities		
5	Execute		
6	Check first activity		
7	Configure software		
8	Test proper system functionality		
9a1	Execute arbitrary repair activity	Error is detected	
9b1	Finish repair	else	

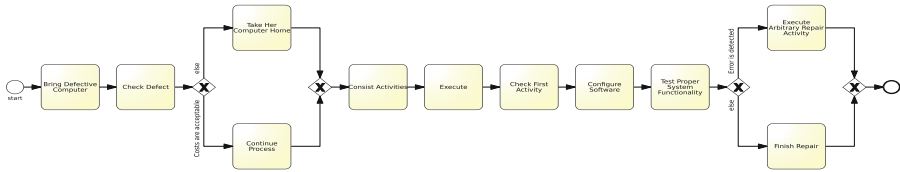


Fig. 6. BPMN process model generated from spreadsheet-based process representation presented in Table 5

For the presented description, our implementation of the algorithm described in Sect. 3 provided the intermediate process model in the spreadsheet-based representation as presented in Table 5 (Fig. 6).

5 Concluding Remarks

In the paper, we present an effective transformation from a semi-formal or informal document into a process model. Such a model can serve as a prototype business process model for further refinement, extension or development.

The proposed solution is based on syntactic analysis of business process description and extracting Subject-Verb-Object constructs, which can be later transformed into process elements. The presented method consists in five steps:

1. Participants extraction.
2. Subject-verb-object constructs extraction.
3. Gateway keywords search.
4. Intermediate process model generation.
5. BPMN diagram generation.

The advantage of our method is the semantic analysis of the text as well as the usage of the intermediate representation, which allows for quick check of the transformation result. Then, the manual correction can be made if necessary. Moreover, as a result we obtain the structured BPMN model, what can help in checking its correctness.

The proposed method of generating process model from natural language description provides some basic information about the described process in the form of BPMN diagram. It is not able to extract more complex constructs and is only able to handle basic elements of BPMN standard. Thus, future works will be focused on enhancing the process models, generated using the proposed method, with additional BPMN elements (such as intermediate events, pools and lanes) as well as adding anaphora resolution to identify real actors in the process. Moreover, we plan to exploit a dedicated domain ontology for exploring related business concepts [15], support verification of the model [16], as well as extend the method to support rules linked to elements of the process [17].

References

1. OMG: Business Process Model and Notation (BPMN) Version 2.0. Technical report, Object Management Group (OMG) (2011)
2. Ferreira, R.C.B., Thom, L.H., Fantinato, M.: A semi-automatic approach to identify business process elements in natural language texts. In: Proceedings of the 19th International Conference on Enterprise Information Systems (2017, to appear)
3. Riefer, M., Ternis, S.F., Thaler, T.: Mining process models from natural language text: a state-of-the-art analysis. Multikonferenz Wirtschaftsinformatik (MKWI 2016), pp. 9–11, March 2016
4. Kluza, K., Honkisz, K.: From SBVR to BPMN and DMN models. Proposal of translation from rules to process and decision models. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2016. LNCS (LNAI), vol. 9693, pp. 453–462. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-39384-1_39
5. Ghose, A., Koliadis, G., Chueng, A.: Process discovery from model and text artefacts. In: 2007 IEEE Congress on Services, pp. 167–174, July 2007
6. de Almeida Rodrigues Goncalves, J.C., Santoro, F.M., Baiao, F.A.: Business process mining from group stories. In: 13th International Conference on Computer Supported Cooperative Work in Design, CSCWD 2009, pp. 161–166. IEEE (2009)
7. Friedrich, F., Mendling, J., Puhlmann, F.: Process model generation from natural language text. In: Mouratidis, H., Rolland, C. (eds.) CAiSE 2011. LNCS, vol. 6741, pp. 482–496. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-21640-4_36
8. Kluza, K., Wiśniewski, P., Jobczyk, K., Ligeza, A., Suchenia Mroczek, A.: Comparison of selected modeling notations for process, decision and system modeling. In: FedCSIS 2017, pp. 1095–1098. IEEE (2017)
9. Yue, T., Briand, L.C., Labiche, Y.: An automated approach to transform use cases into activity diagrams. In: Kühne, T., Selic, B., Gervais, M.-P., Terrier, F. (eds.) ECMFA 2010. LNCS, vol. 6138, pp. 337–353. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13595-8_26

10. Njonko, P.B.F., El Abed, W.: From natural language business requirements to executable models via SBVR. In: 2012 International Conference on Systems and Informatics (ICSAI), pp. 2453–2457 (2012)
11. Ferreira, R.C.B., Thom, L.H., de Oliveira, J.P.M., Avila, D.T., dos Santos, R.I., Fantinato, M.: Assisting process modeling by identifying business process elements in natural language texts. In: de Cesare, S., Frank, U. (eds.) ER 2017. LNCS, vol. 10651, pp. 154–163. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-70625-2_15
12. Sokolov, K., Timofeev, D., Samochadin, A.: Process extraction from texts using semantic unification. In: KMIS, pp. 254–259 (2015)
13. Kluza, K., Wiśniewski, P.: Spreadsheet-based business process modeling. In: FedCSIS 2016, pp. 1355–1358. IEEE (2016)
14. Muehlen, M., Recker, J.: How much language is enough? Theoretical and practical use of the business process modeling notation. In: Bellahsene, Z., Léonard, M. (eds.) CAiSE 2008. LNCS, vol. 5074, pp. 465–479. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-69534-9_35
15. Nalepa, G., Slazynski, M., Kutt, K., Kucharska, E., Luszpaj, A.: Unifying business concepts for SMEs with Prosecco ontology. In: FedCSIS 2015, pp. 1321–1326 (2015)
16. Klimek, R.: A system for deduction-based formal verification of workflow-oriented software models. *Int. J. Appl. Math. Comput. Sci.* **24**(4), 941–956 (2014)
17. Wang, W., Indulska, M., Sadiq, S.: Guidelines for business rule modeling decisions. *J. Comput. Inf. Syst.* **58**(4), 363–373 (2018)



A Study on Performance Sensitivity to Data Sparsity for Automated Essay Scoring

Yanhua Ran, Ben He^(✉), and Jungang Xu

School of Computer and Control Engineering,
University of Chinese Academy of Sciences, Beijing 101408, China
ranyanhua16@mails.ucas.ac.cn, {benhe,xujg}@ucas.ac.cn

Abstract. Automated essay scoring (AES) attempts to rate essays automatically using machine learning and natural language processing techniques, hoping to dramatically reduce the manual efforts involved. Given a target prompt and a set of essays (for the target prompt) to rate, established AES algorithms are mostly prompt-dependent, thereby heavily relying on labeled essays for the particular target prompt as training data, making the availability and the completeness of the labeled essays essential for an AES model to perform. In aware of this, this paper sets out to investigate the impact of data sparsity on the effectiveness of several state-of-the-art AES models. Specifically, on the publicly available ASAP dataset, the effectiveness of different AES algorithms is compared relative to different levels of data completeness, which are simulated with random sampling. To this end, we show that the classical RankSVM and KNN models are more robust to the data sparsity, compared with the end-to-end deep neural network models, but the latter leads to better performance after being trained on sufficient data.

Keywords: Automated essay scoring · Data sparsity
Deep neural network

1 Introduction

Automated essay scoring (AES) is usually considered as a machine learning problem [3, 7, 20] where learning algorithms such as k-nearest neighbor (KNN) and support vector machines for ranking (RankSVM) are applied to learn a rating model for a given essay prompt, after being trained on a set of labeled essays rated by human assessors [5]. Currently, the AES systems have been widely used in large-scale English writing tests, e.g. Graduate Record Examination (GRE), to reduce the human efforts in the writing assessments.

Existing AES systems rely on handcrafted features which encode intuitive dimensions of semantics or writing quality, including lexical complexity, grammar errors, syntactic complexity, organization and development, and coherence etc. [21]. Such AES systems are mostly prompt-dependent in the sense that

they can function well only on the essays for the prompt on which manually labeled essays are available for training. Such prompt-dependent natures sometimes make it hard to apply AES in reality especially when limited rated essays, if any, are available for training. For example, in a writing test, students are asked to write essays for a target prompt without any rated examples, where the prompt-dependent methods are unlikely to perform well due to the lack of training data. Actually, as suggested by a recent study on the task-independent features, the lack of prompt-specific training data is one of the major causes for the degrading performance of current AES methods [23], highlighting the importance of the completeness of training data for AES models. As far as our knowledge, however, it is still not clear to what extent the incompleteness could affect the performance of an AES model and, more importantly, what is the prerequisites for an AES model to function in terms of the number of training data. To mitigate this gap, this paper attempts to investigate the influences of the incompleteness of training data over several AES models by answering following research questions.

- RQ1: How the data sparsity problem affects the performance of different AES methods?
- RQ2: To make an AES model perform, how many rated essays are requested at least for training?

By answering the above questions, we hope to understand the reliances on the completeness of training data in different models and attempt to estimate the least manual workloads that required to make an AES system to perform. To this end, extensive empirical experiments are conducted on the standard Automated Student Assessment Prize (ASAP) dataset¹. As shown by the results, the classical RankSVM and KNN models can learn effective AES models with as few as 20 labeled essays, and they tend to converge when being trained on 200 rated essays. Meanwhile, the AES models based on deep neural network normally require more training data, but could outperform the classical models right after being trained on enough labeled data.

2 Related Work

2.1 Automated Essay Scoring Algorithms

Most of existed Automated Essay Scoring (AES) algorithms view automated essay scoring as classification or regression problem [3, 12–15]. They usually directly learn a classification or regression model based on hand crafted features, such as lexical and syntactic features, and estimate the score of an unseen essay with the prediction of the learned model. In addition, there are also works seeing AES as a ranking problem by applying pairwise learning to ranking algorithms on AES problem [5, 22]. Instead of directly utilizing the prediction of learned

¹ <https://www.kaggle.com/c/asap-aes>.

model as the estimation of an unseen essay as classification or regression, they transform the ranking score given by learning to rank model into the estimation score of the unseen essay by heuristic or learning methods. Intuitively, AES algorithms based on learn to rank take the relative writing quantity between a given pair of essays compared to algorithms based on classification or regression. Experimental results in [5, 22] also show the improvement of learning to rank approaches over traditional classification and regression algorithms. Specifically, Chen and He propose to incorporate the evaluation metric in AES into the loss function to directly optimize the agreement between human and machine raters [4].

Traditional AES models require much work on feature engineering to be effective. Recent years there have been efforts in developing AES approaches based on deep neural networks (DNN), for which feature engineering is no longer required. Alikaniotis et al. propose to learn score-specific word embeddings and utilize a two-layer bi-directional Long-Short Term Memory networks (bi-LSTM) followed by several dense layers to predict essay score [1]. Taghipour and Ng explore a variety of neural network model architectures based on recurrent neural networks [17]. Tay et al. further extend [17] by incorporating neural coherence features [18]. Some works first utilize CNN layers to learn representations of sentences and then CNN [8] or RNN [9] layers are applied to further learn representations of essays.

There are also some works working on the adaption problems between different essay prompts. Phandi et al. propose to apply correlated linear regression to solve the domain adaption problem in AES [14]. They train their AES model using the data from source prompt and a few target prompt essays, such as 10, 25, 50, 100 target prompt essays. Cummins et al. use a constrained multi-task pairwise preference learning approach to combine the data from different prompts to improve the performance [6]. They also find a few target prompt essays is needed to obtain effective results in terms of kappa, a prime evaluation metric for AES.

2.2 Pre-defined Essay Features

In this section, we introduce the hand crafted features used in this work for traditional AES models, such as RankSVM and KNN. These features are widely used in previous works [3, 4, 10, 16] and are concluded in Table 1. The detailed description of the pre-defined hand crafted features are as follows:

Lexical Features

- *Statistics of word length*: The number of words with length in characters larger than 4, 6, 8, 10, 12 in each essay respectively. The *mean* and *variance* of word length in characters in each essay. In general, hard words are likely much longer in length. Statistics of word length are expected to reflect one’s grasp of complex words.

Table 1. Hand-crafted features.

No.	Feature
1	Mean and variance word length in characters
2	Mean length of clauses
3	Essay length in characters and words
4	Number of spelling errors
5	The number of prepositions and commas
6	Mean number of clauses per sentence
7	Mean and variance of sentence length in words
8	Maximum number of clauses of a sentence
9	Semantic vector similarity based on <i>LSA</i>
10	Mean cosine similarity of word vectors by <i>tf-idf</i>
11	The average height of the parser tree of each sentence in an essay
12	Word bigram/trigram frequency <i>tf</i> divided by collection frequency <i>TF</i>
13	POS bigram/trigram frequency <i>tf</i> divided by collection frequency <i>TF</i>

- *Unique words*: The number of unique words in each essay, normalized by the essay length in words. This is expected to reflect a student’s quantity of vocabulary.
- *Grammatical/Spelling errors*: The number of grammatical or spelling errors in each essay. An essay with too many grammatical or spelling errors usually hints a bad grasp of word spelling.

Syntactical Features

- *Statistics of sentence length*: The number of sentences whose length in words are larger than 10, 18 and 25 respectively. The *mean* and *variance* of sentence length in words. The variety of the length of sentences potentially reflects the complexity of syntactics.
- *Clauses*: The *mean* number of clauses in each sentence, normalized by the number of sentence in an essay. The *maximum* number of clauses of a sentence in an essay.
- *Sentence structure*: The average height of the parser tree of each sentence in an essay. The average of the sum of the depth of all nodes in a parser tree of each sentence in an essay.
- *Preposition and Comma*: The number of prepositions and commas in each sentence, normalized by sentence length in words.

Grammar and Fluency Features

- *Word bigram and trigram*: The grammar and fluency of an essay can be measured by the mean tf/TF of n-grams where *tf* is the frequency of n-gram in a single essay and *TF* is the corresponding frequency in the whole essay

collection. A n-gram with high tf/TF generally indicates it is wrong use. We utilize bigram and trigram tf/TF feature in this work.

- *POS bigram and trigram*: Similar with bigram and trigram features, the mean tf/TF of POS bigrams and trigrams are also incorporated.

Content and Prompt-Specific Features

- *Essay length*: The number of words and characters in an essay. Since the fourth root of essay length in words is proved to be highly correlated with the essay score [16], we use the fourth root of the essay length measured by the number of words and characters in experiments.
- *Word vector similarity*: Mean cosine similarity of word vectors, in which the element is the term frequency multiplied by inverse document frequency ($tf-idf$) of each word. It is calculated as the weighted mean of cosine similarities and the weight is set as the corresponding essay score.
- *Semantic vector similarity*: Semantic vectors are generated by Latent Semantic Analysis. The calculation of mean cosine similarity of semantic vectors is the same with word vector similarity.

All the features are normalized through min-max method to avoid several features are dominant compared to other features. For traditional models, it is important to have elaborate feature engineering procedures if we want obtain effective performance. These features are proved to be effective in previous works [3, 4, 10, 16], thus chosen in this work.

3 Data Sparsity Simulation

In this section, we introduce the simulation method of data sparsity in automated essay scoring. We start with a brief introduction to the dataset and the evaluation metric used, followed by the data sparsity simulation method.

Dataset. The dataset used in the experiments is the Automated Student Assessment Prize (ASAP) dataset, the dataset used in the ASAP competition by Kaggle, which is widely used for AES [1, 4, 9]. The statistics of the dataset are summarized in Table 2. There are 8 sets of essays in the dataset from different prompts. All essays are written by students ranging in grade 7 to grade 10. And the essays are different in essay length and score range.

Evaluation Metric. Quadratic weighted Kappa (QWK) is used to measure the agreement between the predicted scores and the corresponding scores from human raters. It is the official evaluation metric in the ASAP competition and is widely used in previous works [1, 4, 22]. QWK is calculated as follows:

$$\kappa = 1 - \frac{\sum_{i,j} w_{i,j} O_{i,j}}{\sum_{i,j} w_{i,j} E_{i,j}}$$

where w , O and E are matrices of weights, observed scores and expected scores. $O_{i,j}$ is the number of essays that receive a score i from the first rater and a

score j from the second rater. $w_{i,j} = (i - j)^2 / (N - 1)^2$, where N is the number of possible scores. E is the outer product between the score vectors of the two raters, normalized to have the same sum as O .

Simulating Data Sparsity. For each of the eight essay sets in ASAP, we divide the dataset into training, validation and testing subsets in line with [1] by utilizing the ids set of validation and test set essays released by them. Concretely, 80% of each essay set are used for training and the remaining 20% for testing. Additionally, 20% of the training data are reserved for validation. The remaining training essays are then deemed a complete training set, out of which essays are sampled to simulate data incompleteness. A simple way of doing so is to randomly select essays from the training set. However, a drawback of this simple simulation would be that the random sample may not reflect the actual score distribution in the complete training set, which in turn biases the training of the AES model. Therefore, in this paper, in order to simulate the data sparsity, the training essays are first sorted by their actual scores, and then partitioned into 5 equal-size bins. From each bin, equal number of essays are randomly sampled as the incomplete training set. In this work, the following training set sizes are sampled to simulate the data sparsity: [5, 10, 15, 20, 25, 30, 50, 100, 150, 200, 300, 400, 500, 600, 700, 800, 900, 1000, 1100]. In consideration of the randomness may lead to large variance especially when training set size is small, the random sampling is repeated 10 time for each sample size, and the average performance trained from the 10 random samples are reported. Compared to the simple simulation that samples from the whole training set, sampling from the sorted bins is expected to preserve the score distribution such that the effect of the random sampling on learned AES model can be minimized.

Table 2. Statistics for the ASAP dataset.

Prompt	#Essays	Grade level	Avg length	Score range
1	1783	8	350	2–12
2	1800	10	350	1–6
3	1726	10	150	0–3
4	1772	10	150	0–3
5	1805	8	150	0–4
6	1800	10	150	0–4
7	1569	7	250	0–30
8	723	10	650	0–60

4 Experimental Setup

In this section, we first introduce the dataset, AES models and evaluation metric. Then we describe the details about how we simulate the data sparsity.

4.1 AES Models

This study uses two classical AES methods based on RankSVM and k-nearest neighbor algorithms, respectively. In addition, a recent state-of-the-art AES model based on end-to-end deep neural networks is also involved in this study. Details of the AES models used are given below.

- **RankSVM** is a pairwise learning to rank algorithm and is applied on AES problem in many works [5, 22]. RankSVM regards AES problem as a ranking problem by taking the relative writing quality for a given essay pair into consideration. First, it assigns a ranking score to each essay in both training and testing essay sets. Then the predicted score of an essay in the testing set is estimated by averaging the human rated score of the K training essays whose ranking scores are nearest the testing essay’s. The value of K is chosen from [6, 8, 10, 12, 14] by maximizing the performance on validation set. The linear kernel RankSVM² is used with the parameter C amongst [3, 4, 5, 6, 7].
- **K nearest neighbors.** K nearest neighbors (KNN) [2] is a simple non-parametric algorithm, of which the main idea is to estimate one’s properties by considering its K nearest neighbors. In this work, we estimate the score of an essay in testing set by averaging the human rated scores of it’s K nearest neighbor essays in training set. The distance is measured by euclidean distance. The value of K is also amongst [4, 6, 8, 10, 12, 14].
- **Neural Model.** Deep Neural network (DNN) is popular in recent years for their good performance on many tasks without feature engineering and has already been successfully applied on AES [1, 9, 17, 18]. In this work, we utilize the architecture of the neural network in [17], which is a state-of-art neural model for AES. First, a LSTM network is applied on the word embedding sequence of a given essay to obtain a list of hidden states. Then these hidden states are averaged and fed into a dense layer with sigmoid activation to get the predicted score, ranging in [0, 1]. The RMSProp optimization algorithm is used to minimized the mean squared error (MSE) loss. The batch size are set to 32. Since the human rated scores are not in [0, 1], we transform the original scores into [0, 1] for training and back into the original range for evaluation. Word embeddings are initialized by the pre-trained embeddings released by [24]. Other setups are also completely in line with [17], we recommend to refer to [17] for space reason.

5 Results and Analysis

In this section, we present the experimental results. The kappa performance of RankSVM, KNN and DNN on incomplete training sets with different sizes are presented in Tables 3, 4 and 5, respectively. As it is widely accepted that the human-machine agreement of an AES system, measured by Kappa, should be

² <http://svmlight.joachims.org/>.

Table 3. The kappa performance of *RankSVM* on incomplete training sets with different sizes.

Size	Prompt Id							
	1	2	3	4	5	6	7	8
5	0.3212	0.2350	0.1793	0.3454	0.2611	0.3292	0.2687	0.2697
10	0.6063	0.4560	0.4141	0.5120	0.5997	0.4808	0.5050	0.4750
15	0.6592	0.5378	0.5041	0.5432	0.6695	0.5933	0.5654	0.5099
20	0.7031	0.5899	0.5736	0.5665	0.7492	0.6524	0.5973	0.5316
25	<i>0.7424</i>	0.6192	0.5533	0.5797	<i>0.749</i>	0.6570	0.6255	0.5674
30	<i>0.7514</i>	0.6457	0.5679	0.5805	<i>0.7613</i>	0.6601	0.6436	0.5524
50	<i>0.7515</i>	0.6522	0.5675	0.6386	<i>0.771</i>	0.6772	0.6609	0.5666
100	<i>0.7613</i>	0.6726	0.5943	0.6613	<i>0.7818</i>	0.7005	0.6974	0.5651
150	<i>0.7815</i>	0.6746	0.6318	0.6617	<i>0.7900</i>	<i>0.7130</i>	0.6981	0.5846
200	<i>0.7848</i>	0.6732	0.6378	0.6715	<i>0.7983</i>	<i>0.7261</i>	0.7120	0.5928
300	<i>0.7983</i>	0.6824	0.6378	0.6811	<i>0.8082</i>	<i>0.7304</i>	<i>0.7297</i>	0.5985
400	<i>0.8006</i>	0.6875	0.6378	0.6856	<i>0.8072</i>	<i>0.7471</i>	<i>0.7349</i>	0.5945
500	<i>0.8009</i>	0.6876	0.6461	0.6868	<i>0.8116</i>	<i>0.7508</i>	<i>0.7394</i>	0.5945
600	<i>0.8041</i>	0.6885	0.6492	0.6908	<i>0.8153</i>	<i>0.7522</i>	<i>0.7442</i>	-
700	<i>0.8125</i>	0.6885	0.6471	0.6945	<i>0.8201</i>	<i>0.7478</i>	<i>0.7469</i>	-
800	<i>0.8137</i>	0.6897	0.6471	0.6939	<i>0.8168</i>	<i>0.7499</i>	<i>0.7416</i>	-
900	<i>0.8137</i>	0.6897	0.6471	0.6963	<i>0.8181</i>	<i>0.7499</i>	<i>0.7412</i>	-
1000	<i>0.8133</i>	0.6895	0.6471	0.6968	<i>0.8191</i>	<i>0.7499</i>	-	-
1100	<i>0.8144</i>	0.6895	0.6520	0.6968	<i>0.8169</i>	<i>0.7499</i>	-	-

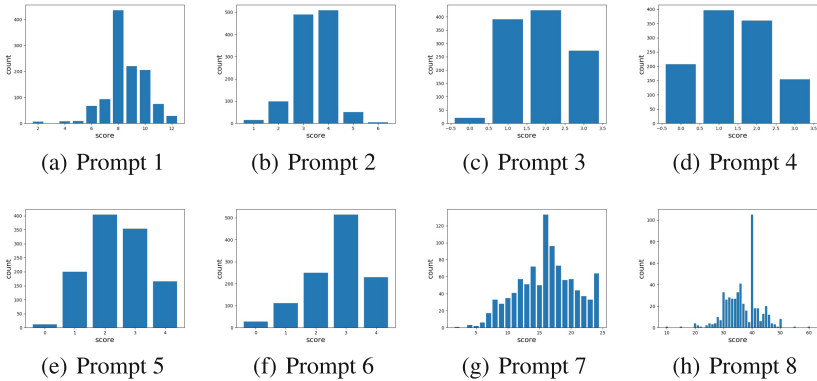
**Fig. 1.** The human rated score distribution of each prompt's training essays.

Table 4. The kappa performance of *KNN* on incomplete training sets with different sizes.

Size	Prompt Id							
	1	2	3	4	5	6	7	8
5	0.2770	0.2374	0.0480	0.2619	0.2842	0.1022	0.1379	0.1649
10	0.5488	0.4603	0.2648	0.5135	0.5674	0.3773	0.3701	0.4085
15	0.5998	0.5266	0.4573	0.5684	0.6614	0.5205	0.4341	0.4480
20	0.6520	0.5451	0.5504	0.5853	0.7056	0.5634	0.4924	0.4458
25	0.6500	0.5580	0.5712	0.6016	<i>0.7192</i>	0.5864	0.5418	0.4782
30	0.6547	0.5737	0.5972	0.6166	<i>0.7222</i>	0.6050	0.5472	0.5040
50	0.6706	0.5754	0.5999	0.6161	<i>0.7373</i>	0.6155	0.5637	0.5481
100	0.7119	0.5877	0.6008	0.6313	<i>0.7616</i>	0.6288	0.5897	0.5789
150	<i>0.7201</i>	0.6205	0.6062	0.6330	<i>0.7660</i>	0.6442	0.6218	0.5969
200	<i>0.7324</i>	0.6432	0.6078	0.6564	<i>0.7656</i>	0.6443	0.6400	0.6216
300	<i>0.7386</i>	0.6523	0.6082	0.6579	<i>0.7678</i>	0.6527	0.6423	0.6177
400	<i>0.7471</i>	0.6505	0.6034	0.6547	<i>0.7695</i>	0.6692	0.6575	0.6190
500	<i>0.7641</i>	0.6508	0.6093	0.6605	<i>0.7802</i>	0.6817	0.6682	0.6162
600	<i>0.7617</i>	0.6531	0.6178	0.6611	<i>0.7812</i>	0.6791	0.6675	-
700	<i>0.7639</i>	0.6535	0.6178	0.6594	<i>0.7812</i>	0.6790	0.6734	-
800	<i>0.7693</i>	0.6535	0.6178	0.6613	<i>0.7825</i>	0.6790	0.6753	-
900	<i>0.7658</i>	0.6591	0.6188	0.6613	<i>0.7825</i>	0.6857	0.6744	-
1000	<i>0.7654</i>	0.6585	0.6184	0.6589	<i>0.7841</i>	0.6881	-	-
1100	<i>0.7821</i>	0.6549	0.6184	0.6589	<i>0.7841</i>	0.6938	-	-

at least 0.70 [19], a training data set is considered sufficient if the learned model results in a performance that meets the 0.70 requirement. Therefore, in the tables, the kappa values larger than 0.70 are in *italic*, and the results improved from below 0.70 are in **bold**. From the experimental results in Tables 3, 4 and 5, we attempt to answer the two main research questions raised in Sect. 1 as follows.

For RQ1, the performance of traditional classification models (RankSVM and KNN) increases fast even if the number of training essays are very small, from 5 to 30. A likely cause for this observation is that the classical RankSVM and KNN models have relatively high tolerance to data sparsity, and can achieve reasonable performance with a small number of training data. Compared to RankSVM and KNN, the performance of the deep neural network model (DNN) appears to be sensitive to data sparsity. When the training set is small, the performance of DNN is much lower than RankSVM and KNN, and the Kappa value is lower than 0.70 on all 8 prompts until the training set size is increased to 200 (see Table 5). We believe this is due to the fact that, compared to RankSVM and KNN, the deep neural network model is complicated, with many parameters to

Table 5. The kappa performance of *DNN* on incomplete training sets with different sizes.

Size	Prompt Id							
	1	2	3	4	5	6	7	8
5	0.1077	0.2116	0.3265	0.3391	0.2611	0.2099	0.2509	0.2173
10	0.2826	0.1861	0.3634	0.4574	0.3172	0.1809	0.3704	0.1581
15	0.2097	0.2604	0.3264	0.5000	0.3909	0.2187	0.3838	0.2568
20	0.2643	0.2537	0.3624	0.5299	0.3283	0.2319	0.3517	0.2551
25	0.2921	0.2576	0.3635	0.5327	0.3973	0.2373	0.4440	0.2507
30	0.2134	0.2061	0.3902	0.4673	0.406	0.2878	0.4020	0.2872
50	0.2764	0.2982	0.4903	0.5462	0.4741	0.3373	0.5146	0.2981
100	0.3294	0.3432	0.5082	0.6191	0.6417	0.4088	0.6079	0.3441
150	0.4274	0.3757	0.5563	0.6411	0.6618	0.5000	0.6696	0.4792
200	0.4537	0.3953	0.6184	0.6803	0.7200	0.5398	0.6993	0.4788
300	0.5371	0.4806	0.6469	0.7183	<i>0.7808</i>	0.6174	0.7194	0.5107
400	0.5746	0.5138	0.6747	<i>0.7508</i>	<i>0.7971</i>	0.6507	<i>0.7595</i>	0.5676
500	0.6075	0.5510	0.6845	<i>0.7590</i>	<i>0.7991</i>	0.6987	<i>0.8029</i>	0.5651
600	0.6172	0.5828	0.6933	<i>0.7647</i>	<i>0.8063</i>	0.7404	<i>0.8106</i>	-
700	0.6615	0.6310	0.6867	<i>0.7615</i>	<i>0.8107</i>	<i>0.7564</i>	<i>0.8189</i>	-
800	0.7740	0.6348	0.6857	<i>0.7828</i>	<i>0.8142</i>	<i>0.7667</i>	<i>0.8254</i>	-
900	<i>0.7780</i>	0.6852	0.6975	<i>0.7809</i>	<i>0.8148</i>	<i>0.7632</i>	<i>0.8281</i>	-
1000	<i>0.7894</i>	0.6774	0.6916	<i>0.7889</i>	<i>0.8225</i>	<i>0.7859</i>	-	-
1100	<i>0.8097</i>	0.6998	0.6945	<i>0.7955</i>	<i>0.8195</i>	<i>0.7870</i>	-	-

learn, and consequently, requires a large amount of training data. On the other hand, the neural network model outperforms the two classical models with the use of all training data available, showing that the deep model has strong ability in learning rating patterns out of sufficient training data.

For RQ2, we find that the Kappa values of RankSVM on prompts 1 and 5 are larger than 0.7 when the training set size is only 20. In addition, RankSVM achieves $\text{Kappa} \geq 0.7$ on prompts 6 and 7 with training set sizes of 100 and 200, respectively. As for KNN, it achieves $\text{Kappa} \geq 0.7$ on prompts 1 and 5 with training set size of 100 and 20. Compared to traditional methods, DNN needs much more training data to surpass the 0.70 threshold, of which the sizes are 800, 300, 200, 600 and 300 for prompts 1, 4, 5, 6, 7, respectively. Overall, the minimal number of training essays required to learn an effective prompt-specific AES model depends on the data and the learning algorithm used. For example, RankSVM requires only 20 training essays to reach the $\text{Kappa} = 0.70$ threshold on prompts 1 and 5, but is unable to outperform this threshold on prompts 2–4 even if all training data available are used. To investigate this issue, we plot the real score distribution of essays for all 8 prompts in Fig. 1. As we can see

Table 6. The performance loss in Kappa in percentage against the use of all training data available.

Size	Model	Prompt Id							
		1	2	3	4	5	6	7	8
30	RankSVM	8.38	6.82	14.8	20.1	7.73	14.0	16.1	8.35
	KNN	19.5	14.9	3.62	7.26	8.57	14.7	23.4	23.3
	DNN	279	239	78.8	70.0	102	173	106	97.6
50	RankSVM	8.37	5.76	14.9	9.11	6.37	11.1	13.0	5.64
	KNN	16.6	14.6	3.16	7.34	6.35	12.7	19.8	13.4
	DNN	192	134	42.3	45.6	73.5	133	60.9	90.4
100	RankSVM	6.97	2.55	9.71	5.36	4.90	7.39	7.10	5.91
	KNN	9.86	12.1	3.00	4.75	2.95	10.3	14.5	7.39
	DNN	145	104	37.3	28.5	28.1	92.5	36.2	65.0
150	RankSVM	4.21	2.25	3.19	5.30	3.81	5.51	7.00	2.39
	KNN	8.60	6.23	2.08	4.48	2.36	7.70	8.60	4.14
	DNN	89.5	86.3	25.4	24.1	24.3	57.4	23.7	18.5
200	RankSVM	3.77	2.46	2.22	3.77	2.73	3.59	4.91	0.970
	KNN	6.78	2.46	1.82	0.750	2.41	7.69	5.52	0
	DNN	78.5	77.0	12.8	16.9	14.2	45.8	18.4	18.6

from this figure, the score distributions of essays written for prompts 1 and 5 are more diverse than those written for prompts 2–4, and it is likely the case that a majority of essays written for prompts 2–4 are of medium quality such that the learned model is unable to differentiate between essays with similar quality. Similar observation can also be made from the score distribution of prompt 8, where most essays receive scores from 30 to 50, and in particular, more than 20% of the training essays (100 out of 500) are rated 40. As a result, it is difficult for the AES model to learn the nuances between essays with very close ratings (e.g. 39 and 40). Note that for some of the prompts, the performance of the AES model can never reach the $\text{kappa} = 0.70$ threshold. We suggest that this is caused by the biased distribution of essays scores, and in this case, more training essays with diverse quality are needed for learning an effective AES model.

To further investigate, we present the performance loss in Kappa, with respect to the use of all training data available, with training data sizes of 30, 50, 100, 150, and 200 essays, in Table 6. When training set sizes are too small, we can find the performances of DNN are far from the corresponding best performance, i.e. the last row in Table 5. From Table 6 we find that when there are only 30 essays in training set, RankSVM is able to achieve good performances on prompts 2 and 5, of which the kappa losses compared to the best performance obtained on much more training essays are only 6.82% and 7.73% respectively. For KNN, the kappa losses on prompts 3, 4 and 5 at training set size 30 are 3.62%, 7.26% and 8.57%, showing that this model converges with only a small number of

training essays. The kappa losses on all eight prompts at training sizes 100 and 200 are within 10% and 5% respectively for RankSVM. This demonstrates that traditional methods are able to achieve relatively stable performance with a few training essays at the cost of small kappa losses, whereas the DNN model requires a large amount of training data to learn, and can indeed lead to the performance that is better than RankSVM and KNN with the availability of enough training data, as shown in the results in Tables 3, 4 and 5.

6 Conclusions

This paper has conducted comprehensive experiments to investigate two key research questions (RQs) regarding the data sparsity problem in automated essay scoring. According to the results, for RQ1, compared to the classical RankSVM and KNN models, the recent deep neural networks are much more sensitive to data sparsity due to its model complexity. For RQ2, in general, the classical models like RankSVM and KNN require a small number of training essays due to their tolerance to data sparsity. They can learn an effective AES model with as few as 20 training essays. Both RankSVM and KNN converge with about 200 training essays at most. The DNN model, in contrast, is relatively data-hungry due to its model complexity. According to the results, the DNN model does not appear to converge even if all training data available are used. Even though, the performance of the AES model learned by DNN is better than RankSVM and KNN in most cases when more than 1,000 training essays are used.

As indicated by the results, in real-life applications, for a given essay prompt, i.e. there are only very few rated essays for training, or even no rated essays at all, it is recommended to use RankSVM for its high tolerance to data sparsity. It is also recommended to rate 20–100 essays by human assessors in order to learn an effective AES model by RankSVM. On the other hand, if there are enough, namely more than 1,000 rated essays available for a given prompt, it is recommended to use neural network model to learn the AES model to fully utilize DNN's power in learning patterns out of sufficient training data. Finally, the results indicate that the performance of prompt-independent AES [11] can be potentially improved by including a small number of labeled essays written for the target prompt.

Acknowledgments. This work is supported by the National Natural Science Foundation of China (61472391).


References

1. Alikaniotis, D., Yannakoudakis, H., Rei, M.: Automatic text scoring using neural networks. In: ACL (1). The Association for Computer Linguistics (2016)
2. Altman, N.S.: An introduction to kernel and nearest-neighbor nonparametric regression. *Am. Stat.* **46**(3), 175–185 (1992)
3. Attali, Y., Burstein, J.: Automated essay scoring with e-rater® v. 2. *J. Technol. Learn. Assess.* **4**(3), 1–31 (2006)

4. Chen, H., He, B.: Automated essay scoring by maximizing human-machine agreement. In: EMNLP, pp. 1741–1752. ACL (2013)
5. Chen, H., Jungang, X., He, B.: Automated essay scoring by capturing relative writing quality. *Comput. J.* **57**(9), 1318–1330 (2014)
6. Cummins, R., Zhang, M., Briscoe, T.: Constrained multi-task learning for automated essay scoring. In: ACL (1), pp. 789–799. The Association for Computer Linguistics (2016)
7. Dikli, S.: An overview of automated scoring of essays. *J. Technol. Learn. Assess.* **5**(1) (2006)
8. Dong, F., Zhang, Y.: Automatic features for essay scoring - an empirical study. In: EMNLP, pp. 1072–1077. The Association for Computational Linguistics (2016)
9. Dong, F., Zhang, Y., Yang, J.: Attention-based recurrent convolutional neural network for automatic essay scoring. In: CoNLL, pp. 153–162. Association for Computational Linguistics (2017)
10. Foltz, P.W., Laham, D., Landauer, T.K.: Automated essay scoring: applications to educational technology. In: World Conference on Educational Multimedia, Hypermedia and Telecommunications, pp. 939–944 (1999)
11. Jin, C., He, B., Hui, K., Sun, L.: TDNN: a two-stage deep neural network for prompt-independent automated essay scoring. In: ACL. The Association for Computer Linguistics (2018)
12. Larkey, L.S.: Automatic essay grading using text categorization techniques. In: SIGIR, pp. 90–95. ACM (1998)
13. Mcnamara, D.S., Crossley, S.A., Roscoe, R.D., Allen, L.K., Dai, J.: A hierarchical classification approach to automated essay scoring. *Assess. Writ.* **23**, 35–59 (2015)
14. Phandi, P., Chai, K.M.A., Ng, H.T.: Flexible domain adaptation for automated essay scoring using correlated linear regression. In: EMNLP, pp. 431–439. The Association for Computational Linguistics (2015)
15. Rudner, L.M.: Automated essay scoring using Bayes’ theorem. *Nat. Counc. Measur. Educ. New Orleans La* **1**(2), 3–21 (2002)
16. Shermis, M.D., Burstein, J. (eds.): *Automated Essay Scoring: A Cross Disciplinary Perspective*. Lawrence Erlbaum Associates, Hillsdale (2003)
17. Taghipour, K., Ng, H.T.: A neural approach to automated essay scoring. In: EMNLP, pp. 1882–1891. The Association for Computational Linguistics (2016)
18. Tay, Y., Phan, M.C., Tuan, L.A., Hui, S.C.: SkipFlow: Incorporating neural coherence features for end-to-end automatic text scoring. *CoRR*, abs/1711.04981 (2017)
19. Williamson, D.M., Xi, X., Jay Breyer, F.: A framework for evaluation and use of automated scoring. *Educ. Measur.: Issues Pract.* **31**(1), 2–13 (2012)
20. Williamson, D.M.: A framework for implementing automated scoring. In: Annual Meeting of the American Educational Research Association and the National Council on Measurement in Education, San Diego, CA (2009)
21. Yang, Y., Buckendahl, C.W., Juszkievicz, P.J., Bholra, D.S.: A review of strategies for validating computer-automated scoring. *Appl. Measur. Educ.* **15**(4), 391–412 (2002)
22. Yannakoudakis, H., Briscoe, T., Medlock, B.: A new dataset and method for automatically grading ESOL texts. In: ACL, pp. 180–189. The Association for Computer Linguistics (2011)
23. Zesch, T., Wojatzki, M., Scholten-Akoun, D.: Task-independent features for automated essay grading. In: BEA@NAACL-HLT, pp. 224–232. The Association for Computer Linguistics (2015)
24. Zou, W.Y., Socher, R., Cer, D.M., Manning, C.D.: Bilingual word embeddings for phrase-based machine translation. In: EMNLP, pp. 1393–1398. ACL (2013)



Extract Knowledge from Web Pages in a Specific Domain

Yihong Lu^(✉) , Shuiyuan Yu, Minyong Shi, and Chunfang Li

School of Computer, Communication University of China, Beijing 100024, China
{yhlu,yusy,myshi,lcf}@cuc.edu.cn

Abstract. Most NLP tasks are based on large, well-organized corpus in general domain, while limited work has been done in specific domain due to the lack of qualified corpus and evaluation dataset. However domain-specific applications are widely needed nowadays. In this paper, we propose a fast and inexpensive, model-assisted method to train a high-quality distributional model from scattered, unconstructed web pages, which can capture knowledge from a specific domain. This approach does not require pre-organized corpus and much human help, and hence works on the specific domain which can't afford the cost of artificially constructed corpus and complex training. We use Word2vec to assist in creating term set and evaluation dataset of embroidery domain. Next, we train a distributional model on filtered search results of term set, and conduct a task-specific tuning via two simple but practical evaluation metrics, word pairs similarity and in-domain terms' coverage. Furthermore, our much-smaller models outperform the word embedding model trained on a large, general corpus in our task. In this work, we demonstrate the effectiveness of our method and hope it can serve as a reference for researchers who extract high-quality knowledge in specific domains.

Keywords: Knowledge extraction · Specific domain · Web corpus
Word2vec

1 Introduction

Word embeddings are shown to boost many NLP tasks like syntactic and semantic word similarity [18], sentiment analysis [16,26], machine translation [19], text classification [13] recent years. However, most research only covers general domain texts and evaluation datasets. In some specific domain, due to the lack of training set and evaluation set, the number of research is limited. However, the applications of domain-specific models are widely needed nowadays. So this paper has two main goals: gather domain-specific training corpus and extract high-quality knowledge from the corpus.

We choose embroidery domain to conduct our experiment and select Word2vec [18–20] as our tool. We manually input some in-domain terms as seed words, then propose a model-assisted method to enrich term set and gather

corpus based on it. Then we extract a distributional model based on observation of two simple but practical metrics. Our much-smaller models outperform the word embedding model trained on a large, general corpus in our task.

As our method is domain-independent, this paper has two main contributions. First, we propose an inexpensive and efficient method to gather qualified specific domain corpus from sporadic web data. Second, we present an approach for gathering and evaluating a distributional model to extract domain-specific knowledge.

The rest of this paper is organized as follows. Section 2 shows some related works on specific domains. A model-assisted method is proposed in Sect. 3. The metrics and the results of the experiment are illustrated in Sect. 4. At last, the conclusion and future work are listed in Sect. 5.

2 Related Works

There have been some research in the specific domains such as medicine [21], psychology [1,2], biomedicine [6]. More generally, some works [5,8,14] investigated how the corpus size and corpus domain influence the performance of the word embedding. However, the studies above are all conducted in popular domains or relied on existing qualified corpus and evaluation datasets.

To address the lack of domain-specific corpus, Pakhomov et al. [21] used freely available PMC biomedical corpus to replace highly restricted clinical data, but it's based on a common term set. Tixier et al. [28] obtained 11M-token corpus from several manually-selected online sources of construction-related texts. Spousta et al. [24] created a billion-token Slovenian corpus by using some catalog of web pages or search engine results for some generic terms as starting points and expand the corpus by web crawlers. But, it may have the trouble of topic shifting.

As to the evaluation datasets, they are mainly organized by the expensive method of manual scoring, like the WordSim-353 [9], SimLex-999 [10], SemEval-2012 [11].

These methods still rely on pre-organized resources or would imply a vast cost to annotate training data, and can't be generalized well.

3 Proposed Method

3.1 Workflow

Our method uses search results for in-domain terms to gather corpus. Thus the most critical step is extracting as many in-domain terms as possible. We take

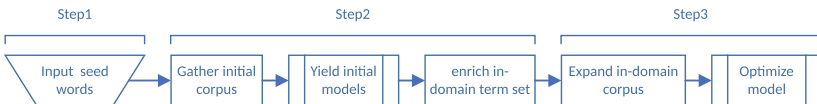


Fig. 1. Flowchart of proposed method

a model-assisted way to extract terms and expand corpus. Specific steps are as follows (Flowchart is shown in Fig. 1):

Step 1. Manually input in-domain terms as seed words. Theoretically, the increasement of diversity and number of seed words will lead to better performance. However, it's difficult to obtain a large number of seed words in specific domain and we haven't found the most appropriate choice of seed words so far. In our task, we select 128 terms and yield a model that performs well enough.

Step 2. Gather initial corpus using search engine results for seed words, remove duplicates and meaningless data, word-tokenize each document, discard stop words, then yield an initial model using Word2vec. After that, enrich the term set with words and phrases that are similar to the seed words with the assistant of the initial model. Finally, we get a 1196-word term set.

Step 3. Enlarge in-domain corpus based on expanded term set and remove irrelevant texts whose similarity with embroidery below a threshold, then use the expanded corpus to optimize the initial model and conduct a task-specific tuning based on observation of evaluation metrics.

3.2 Evaluation Metrics

We design two numerical evaluation metrics for evaluation:

Coverage of In-domain Terms. A good model has to assure good coverage of domain terms. We use the 1196-word term set organized in Sect. 3.1 as evaluation dataset and calculate term coverage by:

$$coverage = \frac{|M \cap D|}{|D|} \quad (1)$$

M represents the set of terms in model and D represents the set of terms in evaluation dataset.

WordSim-Score. We propose a simple but practical evaluation measure which can ease the burden of the annotators. We select 145 pairs of domain synonyms, which should be as close as possible. We use WordSim-Score to evaluate model's capability to capture in-domain term semantics, which is calculated by:

$$WordSim - Score = \frac{\sum_{(a,b) \in word_pairs} (Score(rank_b^a) + Score(rank_a^b))}{2 * len(word_pairs)} \quad (2)$$

$rank_b^a$ represents the similarity of word A and B. If $rank_b^a = 5$, it means that, among all words in the vocabulary, B is the 5th closest word of A. The distribution of Score(x) is shown in Fig. 2.

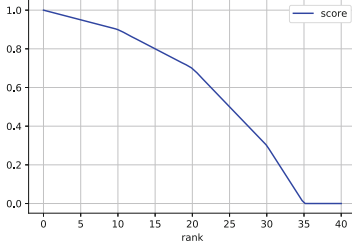


Fig. 2. Distribution of Score(x)

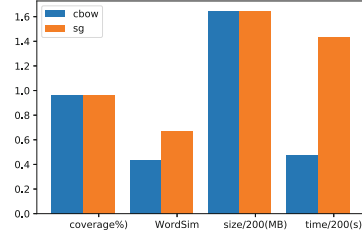


Fig. 3. Effect of different model architectures

4 Experiment

The key factors affecting the quality of the model are input corpora, model architectures, and hyper-parameter settings [6,14], so the following experiments will compare these three factors.

The Choice of Model Architecture. Tasks, domains, and corpus have mixed effects on the selection of models. There is no consistent and comparable view on this topic [4, 6, 14, 18, 19, 27].

In our task (See Fig. 3), the term coverage and model size are similar, but CBOW gets higher WordSim-Score, and only spent a third of the training time equivalent to Skip-gram. Therefore, we select CBOW to perform the following analysis.

Table 1. Statistics of experimental corpus

Model	Description of corpus	Number of tokens
General	Publicly available pre-trained model trained on general domain ^a	65000M
clean0	Raw corpus of embroidery domain	17.5M
clean1	Slightly-cleaned corpus	10.3M
clean2	Deep-cleaned corpus	7M
cl2-ex	Expanded and deep-cleaned corpus	14M
cl2-ex-sf	Shuffled, expanded and deep-cleaned corpus	14M
cl2-wo	Shuffled, expanded and deep-cleaned corpus without treating phrase as one token	14.5M

^a<https://kexue.fm/archives/4304>.

The Choice of Corpus. Which is more important? Size or domain? There is as yet no consensus. In some tasks [7, 14, 23, 25, 28], a model trained on a small domain specific corpus can outperform a model trained on a large but generic

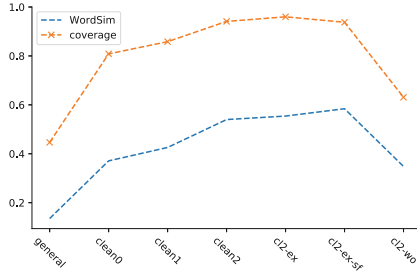


Fig. 4. Effect of different corpora

corpus, while some works [3, 12, 18] show opposite results. Between the two opposing views, some research [6, 17, 22, 29] yields the results that models trained on specific-domain corpora do not necessarily have better performance than those trained on general domain corpora. We study the effect of the general corpus and domain-specific corpus, corpus with three kinds of cleaning degree, and two tips: shuffle sentences and treat phrase as one token. The corpus statistics are shown in Table 1. We summarize the previous study and our experimental results from Fig. 4, and attempt to give some tips for choosing corpus:

1. Ensure the terms of this domain fully trained. If most in-domain terms are included in the general domain, a large general-domain corpus can be used as an alternative. Otherwise, a domain-specific corpus is necessary.
2. When the quantity of corpus is relatively small, the proper cleaning of the corpus (removing low relevancy corpus) can also improve the performance.
3. Treating phrase as one token when pre-processing or training [20] is helpful.
4. Shuffled-corpus perform better as it can avoid the influence of order [6, 13].

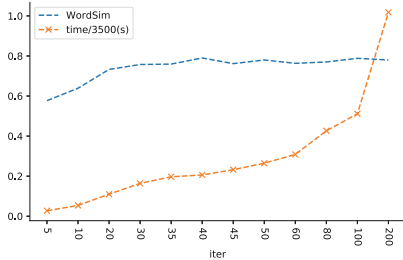


Fig. 5. Evaluation results for iterations (default = 5)

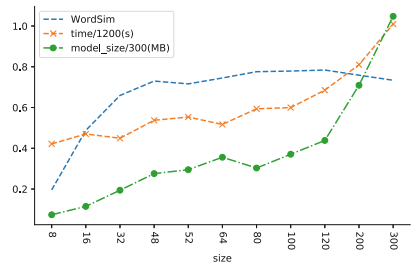


Fig. 6. Evaluation results for vector dimension (default = 100)

The Choice of Hyper-parameters. According to Levy et al. [15], sometimes careful hyper-parameter tuning can even outweigh the importance of adding

more data. As the choice of the hyper-parameter selection is a task-specific decision [14, 20], we conduct a tuning based on observation of WordSim-Score, coverage, model size and training time. It should be noted that we only list parameters and evaluation metrics that contribute to the decision.

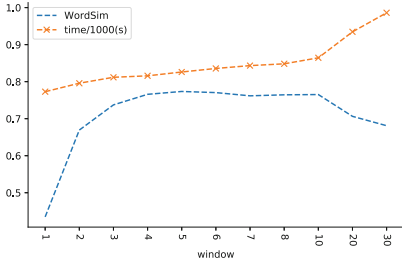


Fig. 7. Evaluation results for window size (default = 5)

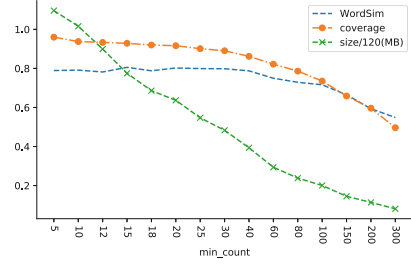


Fig. 8. Evaluation results for min-count (default = 20)

Based on evaluation results, we set the number of iterations to 40 (Fig. 5), the dimensionality of the embedding to 120 (Fig. 6), the context window size to 5 (Fig. 7) and the min-count to 15 (Fig. 8).

5 Conclusion and Future Work

We achieve the goals mentioned in Sect. 1 well. To gather a qualified specific domain corpus, we manually input a few seed words and crawl web corpus to obtain the initial model. Then enrich and clean corpus with the assist of the initial model. To extract high-quality knowledge from the corpus, we design two numerical evaluation metrics for evaluation of distributional models and conduct a task-specific tuning. Our much-smaller models display excellent performance and outperform the large, general model, demonstrating the effectiveness of our method. Our method is inexpensive and domain-independent. Thus it can serve as a reference for researchers who extract domain-specific knowledge in other domains.

We also come across some problem. How to find the most appropriate set of seed words remains a difficult question here. Also, the impact of seed words on final model remains unclear.

In corpus expansion phase, the errors of the initial model will be amplified. In our experiment, there are many shopping-related texts in the initial corpus, so the initial model will believe that commercials and embroidery are highly correlated, which will cause more shopping-related words being added to the expansion corpus. In the future, we may try to use genetic algorithms or other methods to correct errors as early as possible. Meanwhile, some related words with lower frequency are considered to be less relevant due to the lack of training corpus.

References

1. Altszyler, E., Ribeiro, S., Sigman, M., Slezak, D.F.: The interpretation of dream meaning: resolving ambiguity using latent semantic analysis in a small corpus of text. *Conscious. Cogn.* **56**, 178–187 (2017). <https://doi.org/10.1016/j.concog.2017.09.004>
2. Altszyler, E., Sigman, M., Slezak, D.F.: Comparative study of LSA vs Word2Vec embeddings in small corpora: a case study in dreams database. *Science* **8**, 9
3. Altszyler, E., Sigman, M., Slezak, D.F.: Corpus specificity in LSA and Word2Vec: the role of out-of-domain documents. arXiv preprint [arXiv:1712.10054](https://arxiv.org/abs/1712.10054) (2017)
4. Baroni, M., Dinu, G., Kruszewski, G.: Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In: Proceedings of the 52nd Annual Meeting of the ACL, vol. 1: Long Papers, pp. 238–247 (2014)
5. Cardellino, C., Alonso i Alemany, L.: Disjoint semi-supervised Spanish verb sense disambiguation using word embeddings. In: XVIII Simposio Argentino de Inteligencia Artificial (ASAI)-JAIIO 46 (Córdoba, 2017) (2017)
6. Chiu, B., Crichton, G., Korhonen, A., Pyysalo, S.: How to train good word embeddings for biomedical NLP. In: Proceedings of the 15th Workshop on BioNLP. ACL (2016)
7. Diaz, F., Mitra, B., Craswell, N.: Query expansion with locally-trained word embeddings. In: Proceedings of the 54th Annual Meeting of the ACL, vol. 1: Long Papers. ACL (2016)
8. Dusserre, E., Padró, M.: Bigger does not mean better! we prefer specificity. In: IWCS 2017–12th International Conference on Computational Semantics–Short Papers (2017)
9. Finkelstein, L., et al.: Placing search in context: the concept revisited. *ACM Trans. Inf. Syst.* **20**(1), 116–131 (2002). <https://doi.org/10.1145/503104.503110>
10. Hill, F., Reichart, R., Korhonen, A.: SimLex-999: evaluating semantic models with (genuine) similarity estimation. *Comput. Linguist.* **41**(4), 665–695 (2015). <https://doi.org/10.1162/coli.a.00237>
11. Jin, P., Wu, Y.: SemEval-2012 task 4: evaluating Chinese word similarity. In: Proceedings of the First Joint Conference on Lexical and Computational Semantics–Volume 1: Proceedings of the Main Conference and the Shared Task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation, pp. 374–377. ACL (2012)
12. Kusner, M., Sun, Y., Kolkin, N., Weinberger, K.: From word embeddings to document distances. In: International Conference on Machine Learning, pp. 957–966 (2015)
13. Kutuzov, A., Kunilovskaya, M.: Size vs. structure in training corpora for word embedding models: araneum russicum maximum and russian national corpus. In: van der Aalst, W., et al. (eds.) AIST 2017. LNCS, vol. 10716, pp. 47–58. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-73013-4_5
14. Lai, S., Liu, K., He, S., Zhao, J.: How to generate a good word embedding? *IEEE Intell. Syst.* **1** (2017)
15. Levy, O., Goldberg, Y., Dagan, I.: Improving distributional similarity with lessons learned from word embeddings. *TACL* **3**, 211–225 (2015)
16. Maas, A.L., Daly, R.E., Pham, P.T., Huang, D., Ng, A.Y., Potts, C.: Learning word vectors for sentiment analysis. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, vol. 1, pp. 142–150. ACL (2011)

17. Major, V., Surkis, A., Aphinyanaphongs, Y.: Utility of general and specific word embeddings for classifying translational stages of research. arXiv preprint [arXiv:1705.06262](https://arxiv.org/abs/1705.06262) (2017)
18. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint [arXiv:1301.3781](https://arxiv.org/abs/1301.3781) (2013)
19. Mikolov, T., Le, Q.V., Sutskever, I.: Exploiting similarities among languages for machine translation. arXiv preprint [arXiv:1309.4168](https://arxiv.org/abs/1309.4168) (2013)
20. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems, pp. 3111–3119 (2013)
21. Pakhomov, S.V., Finley, G., McEwan, R., Wang, Y., Melton, G.B.: Corpus domain effects on distributional semantic modeling of medical terms. *Bioinformatics* **32**, 3635–3644 (2016). <https://doi.org/10.1093/bioinformatics/btw529>
22. Qu, L., Ferraro, G., Zhou, L., Hou, W., Schneider, N., Baldwin, T.: Big data small data, in domain out-of domain, known word unknown word: the impact of word representations on sequence labelling tasks. In: Proceedings of the Nineteenth Conference on CoNLL. ACL (2015). <https://doi.org/10.18653/v1/k15-1009>
23. Rekabsaz, N., Mitra, B., Lupu, M., Hanbury, A.: Toward incorporation of relevant documents in Word2Vec. arXiv preprint [arXiv:1707.06598](https://arxiv.org/abs/1707.06598) (2017)
24. Spousta, M.: Web as a corpus. In: Zbornik konference WDS, vol. 6, pp. 179–184 (2006)
25. Sugathadasa, K., et al.: Synergistic union of Word2Vec and lexicon for domain specific semantic similarity. In: 2017 IEEE ICIIS. IEEE, December 2017. <https://doi.org/10.1109/iciinfs.2017.8300343>
26. Tang, D., Wei, F., Yang, N., Zhou, M., Liu, T., Qin, B.: Learning sentiment-specific word embedding for twitter sentiment classification. In: Proceedings of the 52nd Annual Meeting of the ACL, vol. 1: Long Papers. ACL (2014). <https://doi.org/10.3115/v1/p14-1146>
27. Muneeb, T.H., Sahu, S., Anand, A.: Evaluating distributed word representations for capturing semantics of biomedical concepts. In: Proceedings of BioNLP 2015. ACL (2015)
28. Tixier, A.J.P., Vazirgiannis, M., Hallowell, M.R.: Word embeddings for the construction domain. arXiv preprint [arXiv:1610.09333](https://arxiv.org/abs/1610.09333) (2016)
29. Wang, Y., et al.: A comparison of word embeddings for the biomedical natural language processing. arXiv preprint [arXiv:1802.00400](https://arxiv.org/abs/1802.00400) (2018)



TCMEF: A TCM Entity Filter Using Less Text

Hualong Zhang¹, Shuzhi Cheng¹, Liting Liu¹,
and Wenxuan Shi¹

Nankai University, Tianjin, China
nankaizhl@gmail.com, shuzhichengspace@163.com,
nkliuliting826@mail.nankai.edu.cn,
shiwx@nankai.edu.cn

Abstract. We often need to cut out a subset of required entities from existing knowledge graphs or websites, when building a knowledge graph for a certain field. In the area of Traditional Chinese Medicine (TCM), we face the task of screening relevant entities from knowledge bases and websites. In this paper, a three-phase TCM entity filter (TCMEF) is proposed, which can identify TCM related entities with high accuracy only using the texts of very short entity titles instead of analyzing the long document texts. The main part of our method is a Short Text LSTM Classifier (STLC), which learns the text style of TCM terms using stroke and character joint features without word segmentation. In addition, an entity representing a person name, which is severe to be classified by STLC, will be picked out by a Person Name Filter (PNF) and further analyzed by a Rich Text Filter (RTF). The filter uses BaiduBaike and HudongBaike (the two largest Chinese encyclopedia websites) as the main data sources. TCMEF gets an F1 score of 0.9275 in classification, which outperforms general word based short text classification algorithms and is close to a Latent Dirichlet Allocation based model (LDA-SVM) using rich texts.

Keywords: TCM entity filter · Short text classification · Chinese stroke

1 Introduction

Extracting entities from the existing open domain knowledge bases, such as DBpedia [1] or CN-DBpedia [2], is a shortcut to start our knowledge graph projects. To our application, it is to take out the Traditional Chinese Medicine (TCM) entities and their pages from BaiduBaike, HudongBaike and YixueBaike. Although these websites provide category labels for entities, missing and inaccurate labels can be easily found. Many TCM herbs like the ‘车钱草’ in BaiduBaike¹ are just labeled as plant, and some TCM clinical terms are merely labeled as medical. So making further entity classification is necessary rather than relying on the existing labels.

Each entity has a title, which is the name we call it. Meanwhile, entities have their detail pages containing properties and rich descriptions, which are so-called rich text. In fact, we can identify whether the entity is a target by recognizing the topic of its rich text. For example, the LDA based short text classification method in [4] can do this task

¹ <https://baike.baidu.com/item/车钱草/8428046>.

well. If we do so, we need to open every entity entrance, which can cause huge computing, storage, and network costs. Therefore, we prefer to pick out target entities by their short titles directly.

Actually some patterns in text style and character shape can be found in TCM titles. The titles of entities are often shorter than 20, many of them are even shorter than 10 characters. It's hard for general short text classification algorithms to collect enough features for these titles. The method we propose not only considers character embedding but also makes use of strokes (the smallest structural units in Chinese). Some meaning of animals, plants and body organs, can be learned by strokes. For example, Chinese characters with component '艹' like the '蒿' in Fig. 3 are often associated with herbs. And the component '艹' can be treated as a sequence of strokes.

In this work, we choose Support Vector Machines (SVM), Classification and Regression Trees (CART), Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM) to test the feature with stroke information. We design a Short Text LSTM Classifier (STLC) as the main part for our TCM Entity Filter (TCMEF) to classify the titles. And we take the performance of a Latent Dirichlet Allocation based algorithm using rich text (LDA_SVM) as a benchmark for it. Our STLC has the same performance as the benchmark. However, some titles of person names are really short and don't follow the style and pattern of TCM. We use a Person Name Filter (PNF) to pick them out and a Rich Text Filter (RTF) to further judge whether they are TCM related.

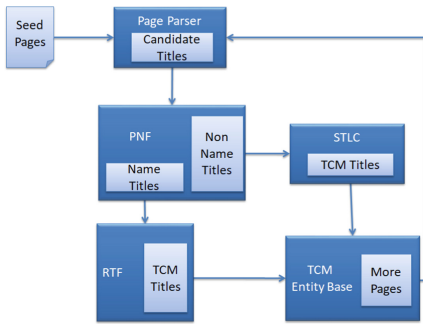


Fig. 1. Framework of TCMEF

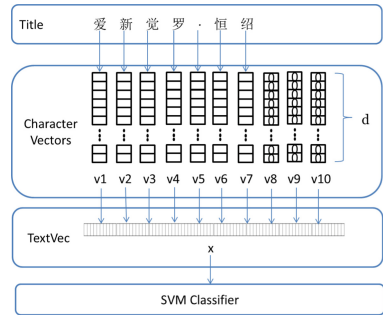


Fig. 2. Workflow of PNF

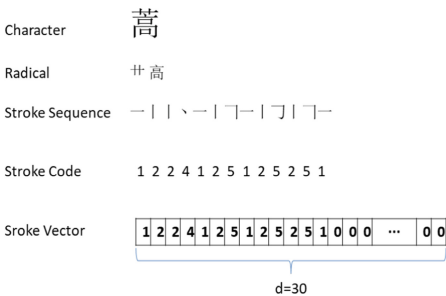


Fig. 3. Stroke encode of character '蒿'

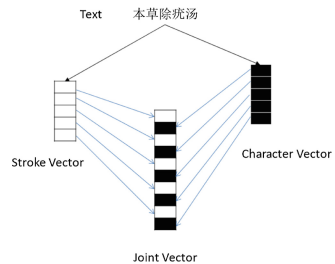


Fig. 4. Merge character and stroke features

2 TCMEF

2.1 Framework and Data Flow

Figure 1 shows the framework of TCMEF and the data flow when it works. Firstly, the titles which are the entries to the entity pages will be picked out by parsing some seed pages. The titles will be sent to PNF for name recognition without jumping to their info pages. A title that is judged not to be a person name will be processed by STLC. STLC is a binary classifier, which extracts features and picks out TCM related titles with high probability. We find it difficult to judge whether a person name has something to do with TCM from its plain text, although some doctors do have TCM style names. So our RTF will open the entity's page and do its topic analysis using the rich text.

2.2 PNF

Chinese names are often less than five characters, and if we use a word segmentation based method, it will be challenging to obtain helpful features. So we need to pick out the person names for further processing. Early work [5] used SVM to identify Chinese names and improve their performance by combining statistical methods. And we take a character embedding when we use SVM to identify person names. Figure 2 provides the workflow of PNF.

Figure 2 depicts that characters in a name title will be converted to 10 d-dimension Character Vectors $\{v_1, v_2, v_3, \dots, v_{10}\}$ since we only use the first ten characters. If the length of the title is shorter than 10, zero vectors fill the space. The Text Vector x to be our input of SVM is a $10 \times d$ dimension vector which we build by concatenating Character Vectors. In practice, considering complexity and performance, 50 is selected as the value of d .

To prevent overfitting, we use a soft-margin linear SVM as our classifier. The objective function is provided by the following equation.

$$\min \frac{1}{2} \|w\|^2 + C \sum_{i=1}^n \xi_i$$

$$s. t. y_i(w^T x_i + b) \geq 1 - \xi_i, \quad i = 1, \dots, n \quad \xi_i \geq 0, \quad i = 1, \dots, n$$

Feed the text vectors x and the label y for training, get the classifier described by weight matrix w and slack variable ξ which is weighted by penalty fact C .

2.3 STLC

STLC is the main part of TCMEF, which is responsible for classifying the text of entity titles to pick out the TCM related ones. It uses a multi-layer LSTM network as a classifier, and its text features involve Chinese character and stroke information.

Stroke Vector. In Chinese, the least symbolic unit is a stroke, not the character. The Chinese character which is called ‘字/Zi’ is a kind of hieroglyphic, and can be regarded as a sequence of strokes as shown in Fig. 3. Some sub-sequences of strokes carry the semantic and pronunciation information of the character. For example, a ‘+’ structure

of the character ‘蒿’ in Fig. 3 indicates that the character is plant-related and is likely to be herbaceous. Chinese characters that contain the same subsequence of strokes are always related in semantics, part-of-speech or pronunciation. We categorize the strokes into five basic types according to the rules shown in Table 1. Then a character can be encoded as a sequence of numbers. For example, in Fig. 3 the Chinese character ‘蒿’ is encoded to 1224125125251. In the 20,902 Chinese characters involved in our work, only 12 rarely used characters have more than 30 strokes. So we use a 30-dim vector to represent the stroke sequence of a Chinese character as shown in Fig. 3.

Table 1. Stroke categorization

code	1	2	3	4	5
stroke	一	丨	丿	丶	㇀ ㇁ ㇂ ㇃ ㇄ ㇅ ㇆ ㇇ ㇈ ㇉ ㇊ ㇋ ㇌ ㇍ ㇎ ㇏

Character Embedding. When using word embedding in Chinese, the performance is highly dependent on the quality of word segmentation. Character embedding can avoid the trouble caused by incorrect word segmentation because we directly encode the characters. Specifically, we get a dictionary matrix D with a size of $d \times N$ that contains all the involved Chinese characters by training with Wiki data Chinese corpus. Where d is the length of each character vector and N is the size of the dictionary. Punctuation marks are also treated as Chinese characters.

Title Representation. After stroke encoding and character embedding, each character will get its stroke vector and character vector. We merge the stroke vectors and character vectors to represent a title. The merge process is shown in Fig. 4. We only use the first 20 characters of a title, because few titles have more than 20 characters.

LSTM. Recurrent neural networks (RNNs) are a family of neural networks designed for sequential data processing. RNNs take a sequence of vectors (x_1, x_2, \dots, x_n) as input and return another sequence (h_1, h_2, \dots, h_n) that represents some state information about the sequence at each step of the input. Due to the vanishing gradient problem, general RNNs can’t preserve distant historical information well [3]. LSTMs incorporate a memory-cell to combat this issue and have shown great capabilities to capture long-range dependencies. An LSTM cell consists of an input gate, an output gate and a forget gate. Both input gate and forget gate are used to update the cell state. Formulated as follows:

$$\begin{aligned}
 i_t &= \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) && (\text{input gate}) \\
 f_t &= \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f) && (\text{forget gate}) \\
 c_t &= f_t \odot c_{t-1} + i_t \odot \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) && (\text{cell state}) \\
 o_t &= \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o) && (\text{output gate}) \\
 h_t &= o_t \odot \tanh(c_t) && (\text{output})
 \end{aligned}$$

where σ is the element-wise sigmoid function, \odot is the element-wise product, \mathbf{W} ’s are weight matrices, and \mathbf{b} ’s are biases.

Network and Workflow. The main structure of STLC is a three-layer neural network classifier as presented in Fig. 5. The feature vector of an entity title is first embedded by an input layer and then processed by an LSTM layer. Finally, the decision value is output by a fully connected layer. To prevent overfitting, we set a 50% elimination rate for dropout. The size of fully connected layer’s output is 1. Since Sigmoid is used as the activation function, the output is a decimal between 0 and 1, representing the probability that the title is a positive case.

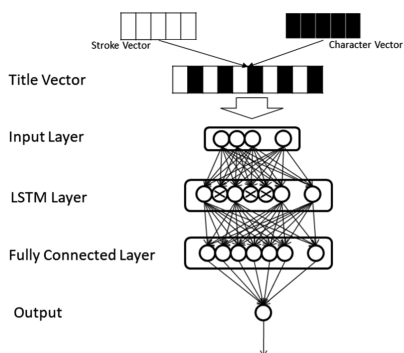


Fig. 5. The network of STLC

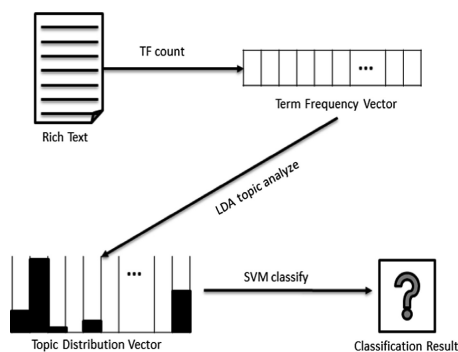


Fig. 6. The workflow of RTF

2.4 RTF

If allowed to classify entities with rich text, studies [6, 7] show that LDA based algorithms work well in both Chinese and English. When a title is judged as a person name by PNF, our LDA-SVM Classifier of RTF will use the rich text to make a further thematic analysis. Figure 6 shows the workflow of RTF.

LDA-SVM Classifier. Blei and Ng [8] use the topics in LDA as features of text for classification. Applying LDA, weighted terms represent topics. Weighted topics in a polynomial distribution represent a document. In this paper, we use the LDA toolkit in sklearn to convert the rich text into a vector which represents the topic distribution of a document as shown in Fig. 6. The length of this vector represents the number of topics we select for LDA. An SVM classifier will be the final judge to judge whether this document is TCM related.

3 Experiment and Analysis

In order to know the performance of TCMEF in TCM entity classification, we conduct experiments on its three modules separately and also do integration testing. We select several popular short text classification algorithms as control groups for STCL. And we take an LDA based document classification algorithm as a benchmark. F1 is selected as our evaluation.

Table 2. Dataset categories

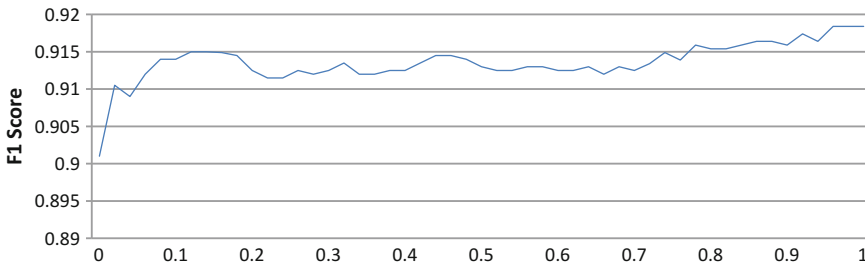
Label	Category	Count	Example
Positive	TCM Basic Terms	1000	过逸凉燥, 火易动血, 迂正, 形气
	Medical Books	600	医方论, 伤寒杂病论, 《女丹合编》
	Prescription	1000	丁矾散, 丁香果散, 麦味地黄汤
	Medicinal Materials	2000	大山玄参, 凤眼兰, 串铃草, 千叶独活根
	Therapy	500	温经通阳, 润燥化痰, 调理肠胃, 解表
	Disease and Symptom	1000	血虚自汗, 肾虚自汗, 肾亏, 喉痹
	Zhenghou	800	肝郁化火, 气血瘀滞鼻窍证, 邪扰胸膈
Negative	Person Name	500	华佗, 张浩良, 黄吉康, 夏应堂
	Random Title	12000	亚洲锂都, 天敦咨询有限公司, 百盛绿盾380, 土笛
	Person Name	1000	冯希望, 连小明, 陈洁仪, 高晓松

3.1 Datasets

We select 7500 TCM related entity titles manually from two largest Chinese knowledge bases Baidu Baike and Hudong Baike. In addition, we randomly select 13,000 open domain titles as negative samples for supervised learning. These data are mainly used for the classification experiments of STLC and RTF. Table 2 lists the detailed categories of these data. The 1000 extra person names in the negative group are used to see whether person names have a negative effect on classification. For PNF, we randomly select 5000 Chinese names from a name corpus for training. For rich text filtering, we parse entity pages from Baidu Baike and Hudong Baike.

3.2 Person Name Filtering

We prepare 5000 positive samples and the same number of negative samples for the name filtering experiment. 20% of them are used as the test data, the rest for model training. The core model we select for PNF is a soft-margin SVM classifier. Non-linear kernel functions are proved to lead to overfitting in our practice. Experimental results in Fig. 7 show that low penalty fact C lead to poor performance. Therefore, we finally set C to 1 and use no kernel. The F1score of PNF reaches 0.9184.

**Fig. 7.** Performance test on SVM penalty fact C

3.3 Short Text TCM Title Classification

The data used for TCM short text classification is listed in Table 2. 80% of them are used for model training and the rest for classification test. The dimension d of character vectors is set to 50 considering the complexity and performance. Table 3 shows the performance of STLC and other algorithms on the given task. We choose popular algorithms including LSTM, SVM and CART as a classifier to see whether the model achieves better performance by introducing stroke features.

Removing the case of person names from the dataset, we can get better classification results as Table 4 conveyed. Applying LSTM using the character-stroke feature on titles with no person name, which is the main job of STLC, we can get an F1 score of 0.9257. This also proves that it makes sense to remove person names before short text classification.

Table 3. Result of short text TCM title classification

Feature	LSTM	SVM-linear	CART	RNN
word_seg	0.8305	0.8353	–	0.8745
character	0.8952	0.7886	0.7788	0.8663
stroke_seq	0.7871	0.7123	0.7281	0.6948
character & stroke_seq	0.9036	0.8346	0.7948	0.8785

Table 4. Classification performance on full dataset and no-person name dataset

	Full dataset	No person name
LSTM	0.9036	0.9257
SVM-linear	0.8346	0.8403
CART	0.7948	0.8086
RNN	0.8785	0.8969

3.4 Rich Text Filtering

The task of rich text filtering is done by an LDA-SVM model. We do topic learning and classification using all information of entity pages, so the representation of the document involves pre-work such as text extraction, word segmentation, stop words removing and vectorization. It can be seen from Table 5 that a good result can be obtained when the topic number is around 180. The parameter $n_feature$ represents the number of top feature words for representation of each document.

3.5 TCMEF Integrated Experiment

For a rich text classifier, getting an F1 score of 0.92 or more is not difficult. We hope our TCMEF will achieve comparable performance using short titles and very limited rich text. The integration experiment involves all the data listed in Table 2. The raw data is divided into two parts by a pre-trained PNF. RTF handles the suspected person names and STLC handles the rest. They work on their data with a 4:1 train-test ratio.

The performance report is displayed in Table 6. STLC and RTF both performed well in their tasks. The overall performance after integration also reaches our expectation with an F1 score of 0.9275. And we see that thanks to the work of PNF, we only have to download pages and rich text for person name entities. In open domain knowledge bases, the proportion of person names is often less than 10%.

Table 5. Experiment result of RTF

Topic number	n_feature				
	200	400	600	800	1000
40	0.8662	0.8104	0.8506	0.8476	0.8576
60	0.8443	0.8558	0.8231	0.8993	0.8894
80	0.8785	0.8766	0.874	0.9246	0.9171
100	0.9043	0.8866	0.9145	0.9045	0.8993
140	0.9094	0.9121	0.897	0.8941	0.9172
180	0.9247	0.8811	0.8992	0.9247	0.9449
200	0.9221	0.9068	0.8888	0.9247	0.9248

Table 6. Performance of integration

	Precision	Recall	F1	Support
STLC	0.9271	0.9273	0.9271	3245
RTF	0.93	0.9298	0.9284	527
TCMEF	0.9276	0.9275	0.9275	3772

4 Conclusion

In this paper, we propose a three-phase filtering framework for TCM entities. It uses the stroke features of Chinese characters to recognize the TCM short text titles and make individual identification for person names. It uses only short texts with an average length of less than 20 and insufficient rich text to work, with performance close to the LDA-SVM model using all rich text. It saves network, storage and computing costs for online entity filtering since we only have to download pages and analyze rich text for less than 10% of the entities.

References

1. Bizer, C., Cyganiak, R.: A nucleus for a web of open data. In: Conference, ISWC 2007 + ASWC 2007, Busan, Korea, November, pp. 722–735. DBLP (2007)
2. Xu, B., et al.: CN-DBpedia: a never-ending chinese knowledge extraction system. In: Benferhat, S., Tabia, K., Ali, M. (eds.) IEA/AIE 2017. LNCS (LNAI), vol. 10351, pp. 428–438. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-60045-1_44

3. Bengio, Y., Ducharme, R., Vincent, V., Jauvin, C.: A neural probabilistic language model. *J. Mach. Learn. Res.* **3**(Feb), 1137–1155 (2003)
4. Sun, B., Zhao, P.: Feature extension for Chinese short text classification based on topical N-Grams. In: *International Conference on Computer and Information Science*
5. Mao, T.T., Li-Shuang, L.I., Huang, D.G.: Recognizing Chinese person names based on hybrid models. *J. Chin. Inf. Process.* **21**(2), 22–28 (2007)
6. Lee, S., Baker, J., Song, J.: An empirical comparison off our text mining methods. *J. Comput. Inf. Syst.* **51**(1), 1–10 (2010)
7. Wu, X., Fang, L., Wang, P.: Performance of using LDA for Chinese news text classification. In: *Electrical and Computer Engineering*, pp. 1260–1264. IEEE (2015)
8. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. *J. Mach. Learn. Res.* **3**, 993–1022 (2003)

Image and Video Data Analysis



Two-Stage Object Detection Based on Deep Pruning for Remote Sensing Image

Shengsheng Wang¹, Meng Wang^{1(✉)}, Xin Zhao¹, and Dong Liu²

¹ College of Computer Science and Technology,
Jilin University, Changchun 130012, China
1030997649@qq.com

² Xiangnan University, Chenzhou, China

Abstract. In this paper, we concentrate on tackling the problems of object detection in very-high-resolution (VHR) remote sensing images. The main challenges of object detection in VHR remote sensing images are: (1) VHR images are usually too large and it will consume too much time when locating objects; (2) high false alarm because background dominate and is complex in VHR images. To address the above challenges, a new method is proposed to build two-stage object detection model. Our proposed method can be divided into two processes: (1) we use twice pruning to get region proposal convolutional neural network which is used to predict region proposals; (2) and we use once pruning to get classification convolutional neural network which is used to analyze the result of the first stage and output the class labels of proposals. The experimental results show that the proposed method has high precision and is significantly faster than the state-of-the-art methods on NWPU VHR-10 remote sensing dataset.

Keywords: Very-high-resolution remote sensing image · Computer vision
Object detection · Convolutional neural network · Deep learning

1 Introduction

The spatial resolution of optical remote sensing sensor has been greatly improved in the past 10 years, and a large number high-resolution images have been applied to resource survey, natural hazard, urban traffic control and other fields [1–7]. The background of remote sensing image is more complex than ground-based image and remote sensing images often contain a lot of noise. Remote sensing images have its own unique characteristics like, providing a vast visual field, covering large area, more visualized contents. The traditional object detection methods based on SIFT [8] and HOG [9] are not ideal for processing such complex images.

In recent years, deep learning began to replace the traditional image processing technology in many computer vision tasks such as object detection, classification. And the two-stage object detection method based on deep learning has been applied to detect objects from high resolution remote sensing image [10–14] recently. “Two-stage” means the region proposal stage [15] and the classification stage. The region proposal stage should extract region proposals rapidly because remote sensing images

are too large, and the classification stage should classify region proposals accurately because the background is complex.

Long et al. [10] first used Selective Search (SS) algorithm to generate region proposals, and then multiple targets were detected by CNN (Convolutional Neural Network) and SVM classifier for optical remote sensing images. Jiang et al. [11] proposed a method that first got proposals by a graph-based super pixel segmentation, and then classified proposals with a CNN. Ševo et al. [12] cut high-resolution satellite images into same size image patches directly and classified the image patches by a trained CNN. Wu et al. [13] used Edge Boxes to extract region proposals and input region proposals to CNN for classification. However, methods above have drawbacks: using SS and super-pixel segmentation to extract region proposals is slow; Cutting image into same size patches will loss image information; it is hard for Edge Boxes to handle images with complex background. In conclusion, the existing two-stage object detection methods based on CNN have the problems of high false alarm rate and slow speed when facing high-resolution remote sensing images.

To solve the above problems, we propose a two-stage Object Detection based on Deep Pruning (ODDP) for VRH remote sensing images. Firstly, a deep neural network pruning method Deep Pruning (DP) is proposed, and then the Learning Region Proposal Network algorithm (LRPN) is proposed based on DP. We use LRPN to train a highly sparse CNN to extract region proposals quickly. Finally, the Optimizing Classification Network algorithm (OCN) based on Deep Pruning is proposed, which is used to learn a more accurate classification network than normal training network. Combine the two networks to obtain the object detection model, and Fig. 1 is the test phase of the object detection model.

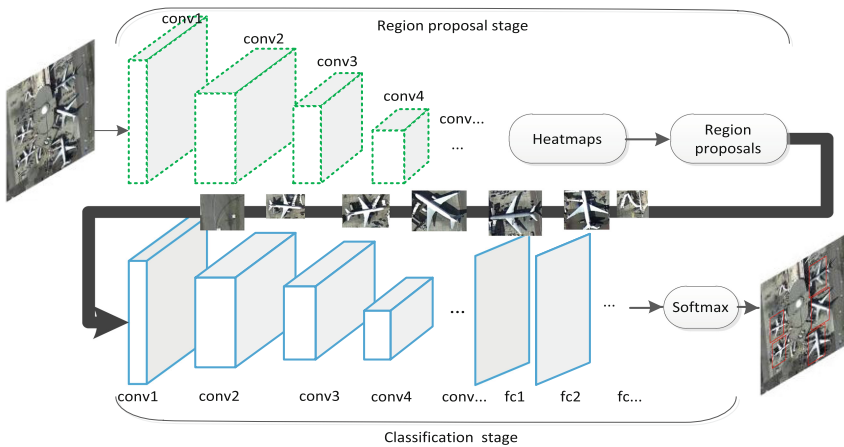


Fig. 1. Test phase of the two-stage model obtained by ODDP.

2 Two-Stage Object Detection Based on Deep Pruning Method

The proposed ODDP includes LRPN and OCN. As can be seen from Fig. 2, LRPN obtains region proposal network through four steps: pretraining, first pruning, reconstruction and secondary pruning. OCN uses the pretraining network to obtain the classification network through pruning and recovering. We construct object detection model by combining region proposal network and classification network.

2.1 Deep Pruning

Deep neural network is typically over-parameterized, and there is a significant redundancy [16], which may lead to overfitting and calculating slowly. Inspired by dropout method of neural network, Deep Pruning (DP) is proposed to reduce the network capacity. Specifically, the proposed method cuts off the connections below the adjustable threshold, which changes the structure of the network, and makes the network sparser. So, DP makes the network less likely to overfitting and accelerates network computing.

There are four steps of DP: (1) Make the network learn which connections are more important. (2) Set threshold, cut off unimportant connections, and then get a sparse network. (3) Train the sparse network to restore the precision. (4) Analyze the average precision and threshold, then choose a way by the result of C. We will describe the workflow of DP below.

Sparsity is the percentage of connections that will be cut off. The processes of choosing threshold are as follows: if there are M parameters in layer W, we sort the M parameters and then choose a parameter as the threshold T:

$$\begin{aligned} T &= \text{sort}(M)_{(1-S) \cdot \text{size}(M)}, \text{ where} \\ S &= S + \text{sig}(\Delta P) \cdot (S_{\text{fixed}} - \theta \cdot S_{\text{conv}} - (1 - \theta) \cdot S_{\text{fully}})^2 \end{aligned} \quad (1)$$

We define C here which will be used in the algorithm below:

$$C = \begin{cases} 1, \Delta P < 0 \\ 2, \Delta P \geq 0 \&\& \theta \cdot S_{\text{conv}} + (1 - \theta) \cdot S_{\text{fully}} < S_{\text{fixed}} \&\& T > 0 \\ 3, \Delta P \geq 0 \&\& (\theta \cdot S_{\text{conv}} + (1 - \theta) \cdot S_{\text{fully}} \geq S_{\text{fixed}} \mid T = 0) \end{cases} \quad (2)$$

Where S is the sparsity which will change during two pruning processes, we set the initial value of S to 0.1, sort() is the sort function which is used to sort M parameters, size() is used to compute the number of parameters, sig() is a signal function which will return -1 or +1, S_{fixed} is the preset sparsity and the value of S_{fixed} should be different in different network structure (in our experiment we set $S_{\text{fixed}} = 0.8$), S_{conv} is the convolution layer sparsity of the network, S_{fully} is the fully connected layer sparsity of the network and we can compute S_{conv} and S_{fully} after each pruning, T is the threshold which is used to prune connections, ΔP is the average precision difference of the network before and after pruning, θ balances the contribution of S_{fixed} and S_{conv} to the

sparse degree of the entire neural network and we select θ according to network structure (in our experiment we set $\theta=0.2$).

Algorithm 1: workflow of Deep Pruning

Input: a network; training dataset

Output: a network

Step 1: Train the network by Stochastic Gradient Descent(SGD): SGD(W)

Step 2: Prune

2.1: Sort the weights and find the threshold

2.2: Cut off the connections with the weight less than threshold

Step 3: Train the sparse network: SGD(W)

Step 4: Calculate C in formula (2),

if C == 1 : recover the cut off connections and move to step 1

if C == 2 : move to step 2

if C == 3 : end

2.2 Learning Region Proposal Network

LRPN makes the initial network to autonomously learn a convolutional neural network with new structure from the training dataset. The new structure network has the following advantages: (1) the network structure fits to the distribution of dataset; (2) avoid the blindness of artificially designed network structure; (3) combine the advantage of the classic initial network and the new sparse network. Due to the above advantages, our method speeds up network computing, while the accuracy does not decrease. The LRPN algorithm and its workflow will be described below.

Algorithm 2: workflow of LRPN

Input: initial network, pretraining dataset, training dataset

Output: pretrained network, RPN

Step 1: Pretraining

1.1 Initialize the weight of initial network: $W \sim N(0, 0.01^2)$

1.2 Use Stochastic Gradient Descent(SGD) and the pretraining dataset to train the initial network: SGD(W)

1.3 Output the pretrained network

Step 2: First pruning

Deep Pruning takes as input the training dataset and the pretrained network to get the sparse network.

Step 3: Reconstruction

Combine conv_fc_class and conv_fc_bbr with convolution layer of the spare network.

Step 4: Secondary pruning

4.1 Deep Pruning takes as input the training dataset and the reconstruction network to get the sparser network.

4.2 Take the final network's convolutional network as RPN and then output RPN.

Pretraining. Deep neural networks always have millions of parameters and training so many parameters is a problematic with hundreds of remote sensing images. Fortunately, we can transfer pretrained networks to our task because the low level convolutional kernels extract the similar features. So, this step provides primary feature for the next step to make the network converge faster even if the training set is really small. Pretraining uses the remote sensing dataset AID (Aerial Image Dataset) [17] rather than ImageNet dataset [10, 18] to train the initial network for getting better feature.

First Pruning. We use deep pruning to change the initial network structure. The training dataset is cropped from NWPU VHR-10 remote sensing dataset [18]. We input the training dataset with its class label, and the pretrained network to Deep Pruning to get a spare network.

In the first pruning, we remove the connections that is not important to the binary classification (object/background) and leave space for the localization task by setting a smaller sparsity.

Reconstruction. Reconstruction allows the network to have the ability to handle both binary classification and localization simultaneously. In this step, we first select the convolution layer of the network that previous step output. And then add two branches after the final convolution layer: one is used to distinguish background and object called `cls_fc_class`; another provides coordinate offset named `cls_fc_bbr`. The output of `cls_fc_class` is entered to softmax classifier during training, and the softmax gives the probability of the image as object or background. `cls_fc_bbr` uses L1 loss function in training to perform bounding box regression.

Secondary Pruning. The secondary pruning changes the network structure again. We input the cropped training dataset with its class label and ground truth bounding box, and the reconstruction network to Deep Pruning to get a sparser network. During deep pruning, first of all, train the network so that we can know which connections are effective not only for the binary classification, but also to the localization. Since there are two tasks, the corresponding objective function should be multi-task loss function. Therefore, the multi-task loss function that optimizes the binary classification and localization tasks at the same time is:

$$L_{\text{ODDP}}(loc, p_{\text{bic}}) = L_{\text{bic}}(p_{\text{bic}}) + \alpha L_{\text{bbox}}(loc) \quad (3)$$

Where loc is the predicted tuple for bounding box regression, p_{bic} is the class confidence calculated by the softmax classifier, L_{bic} is the softmax loss for binary classification of object and background, L_{bbox} is smooth L1 loss:

$$L_{\text{bbox}}(loc) = f_{\text{L1}}(loc - loc_t), \text{ where} \quad (4)$$

$$f_{\text{L1}}(x) = \begin{cases} 0.5x^2, & \text{if } |x| < 1 \\ |x| - 0.5, & \text{otherwise} \end{cases}$$

Where loc_t is the ground truth tuple for bounding box regression.

The background is meaningless in the back propagation, but it will cause the model to converge prematurely. Therefore, set the background sample $\alpha = 0$ and the object sample $\alpha = 0.5$.

Then the secondary pruning is performed to remove the redundant connections to the localization and the binary classification tasks. The secondary pruning method and the first pruning both use Deep Pruning method, while the sparsity of the two pruning process are different.

With the above training completed, Region Proposal Network (RPN) is constructed by the final network's convolution layers. So, the RPN is actually a fully convolutional neural network. At test, RPN takes image Gaussian pyramid to obtain heat maps, and we can get central coordinates of proposals from heat map's local maximal positions.

2.3 Optimizing Classification Network

The benefits of complex networks are very expressive and can capture the highly nonlinear relationship between features and output. The drawback of large network is that it tends to capture the noise in the training dataset. This noise cannot be generalized to new datasets, resulting in overfitting, high variance and weak generalization ability. Simply reducing the capacity of the model leads to another extreme that the network will miss the correlation between the features and output, resulting in underfitting and high bias.

In the existing work [10–13], when training classification network, they selected a network based on experience. So the dataset and network capacity do not match well, which may lead to overfitting or underfitting. We propose a training algorithm Optimizing Classification Network algorithm (OCN) to regulate the network, so that the network can better learn the distribution of dataset. The proposed OCN and the workflow are described in detail below.

In order to achieve consistency with the region proposal network, we use the same pretrained network with LRPN. Firstly, fine-tune the pretrained network, and then set a bigger sparsity than first pruning of LRPN to prune the network by Deep Pruning. Compared with region proposal network, classification network needs to learn the unique features of each class, and to distinguish the subtle differences between multiple classes. So, we need a bigger capacity model which has more connections and that is why after pruning and retraining the network, the cut off connections are recovered.

Algorithm 3: workflow of OCN

Input: training dataset, the pretrained network

Output: classification network

Step 1: Deep Pruning takes as input the training dataset and the pretrained network to get the sparse network.

Step 2: Recover the cut off connections and initialize the weights

Step 3: Train the new network SGD(W)

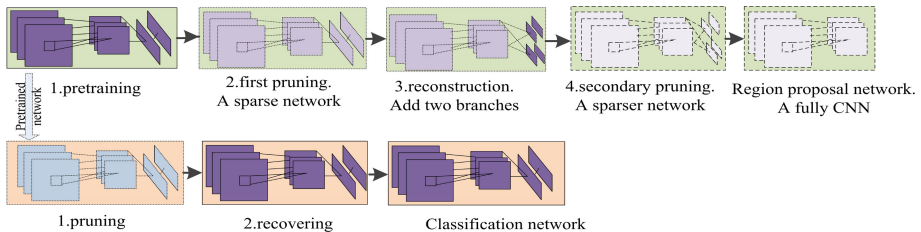


Fig. 2. Framework of ODDP. ODDP includes LRPN and OCN algorithms.

3 Experiment and Analysis

We evaluate the proposed ODDP on a public VHR remote sensing image dataset named NWPU VHR-10 [18], which has 10 classes. We use number 1 to 10 to represent 10 classes during training phase. In this experiment, we only use positive samples of the dataset, which include 650 VHR images. Some of the pictures are from Google maps, and their spatial resolution is between 0.5 m and 2 m. The other part is from Vaihinge dataset, and their spatial resolution is 0.08 m. In our experiment, we select 50% of the whole dataset for training set, 20% for validation set and 30% for test set. To evaluate ODDP, we use average running time, Average Precision (AP) and mean Average Precision (mAP).

We select AlexNet as the initial network and AID as the pretraining dataset. We define image patches as positive or negative samples based on $\text{IoU} \geq 0.7$ or $\text{IoU} < 0.3$. IoU is an evaluation of object detection, which is the overlap rate of two boxes, that is

$$I = \frac{G \cap D}{G \cup D} \quad (5)$$

Where, I is the overlap rate, D (Detection result) is the box predicted by the object detection model, G (Ground truth) is the ground truth box of the detected image.

The hardware environment is 2.8 GHz, 6 core CPU, 32 GB memory, GTX Titan X.

Figure 3 is the detecting result of some images of Vaihinge dataset with 0.08 m spatial resolution. It shows that our proposed method can well detect objects in VHR images. The rectangles in the images are the predicted bounding boxes. The first value in the upper left corner above rectangle is class label, and the second value is the probability that the area within the rectangle belongs to the class. The yellow solid ellipses mark false negative and the green dotted ellipses mark the false positive. It can be seen that although the objects are very different in size, shape and texture, ODDP still detected most objects. We compare ODDP with four current optimal methods, that are COPD [19], transferred CNN [20], RICNN with or without fine-tuning [18]. To be convincing, ODDP and four comparative methods all adopted the same training and test dataset. In addition to COPD, the three comparison methods adopted AlexNet network structure.

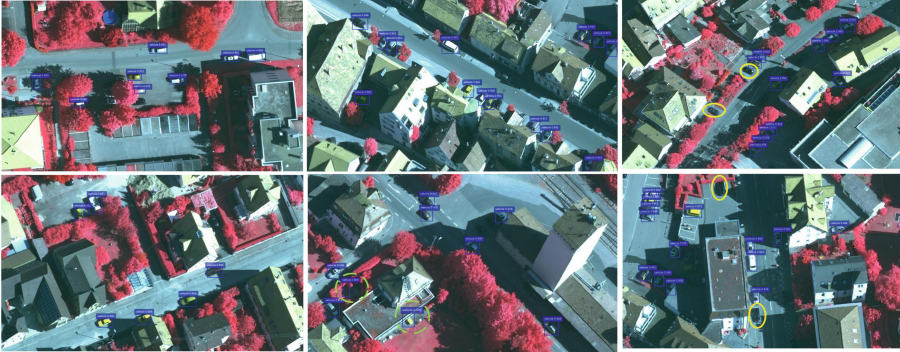


Fig. 3. Object detection results with the proposed method, and the images are from Vaihinge dataset (0.08 m spatial resolution). (Color figure online)

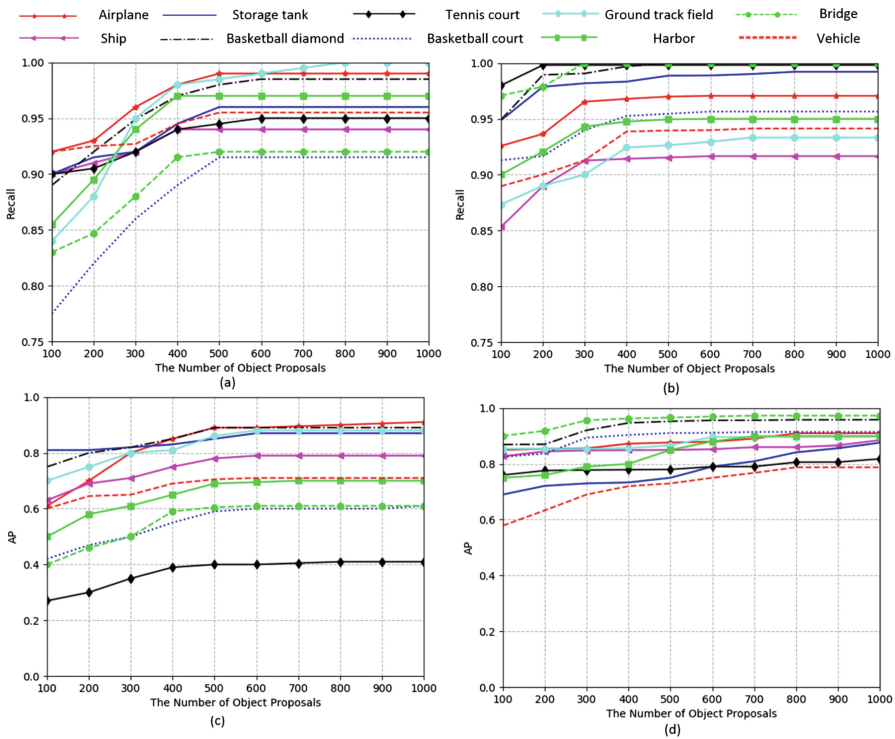


Fig. 4. (a), (b), (c), (d) show the relationship between the hyperparameter (the number of object proposals) and the detecting results (recall and AP). (b), (d) are the detecting results of our proposed method ODDP, and (a), (c) are the results of the RICNN with fine-tuning in [18].

Tables 1, 2 and Fig. 4 are comparison results. From Table 2 we can see that ODDP has better AP than the other four methods. Compared with RICNN with fine-tuning, the average precision of ten classes predicted by the proposed method improve 5.52%,

Table 1. Computation comparisons of five different methods

Algorithm	COPD	Transferred CNN	RICNN without fine-tuning	RICNN with fine-tuning	ODDP
Average running time/s	1.07	5.24	8.77	8.77	0.39

Table 2. The average precision of five object detection methods for 10 classes. The highest score is bold in each row (%)

	COPD	Transferred CNN	RICNN without fine-tuning	RICNN with fine-tuning	ODDP
Airplane	60.44	64.48	84.33	85.39	90.91
Ship	67.85	55.71	74.53	75.38	88.46
Storage tank	60.31	81.38	66.87	84.95	87.53
Baseball diamond	81.20	79.78	85.28	85.69	95.79
Tennis court	52.23	56.46	63.82	65.70	81.79
Basketball court	36.71	46.73	57.18	57.96	91.39
Ground track field	82.71	78.26	82.77	86.32	89.61
Harbor	82.72	78.26	82.79	84.12	89.90
Bridge	17.13	45.97	58.25	61.28	97.27
Vehicle	44.09	43.01	66.56	69.88	78.76
Mean AP	58.54	63.00	72.24	75.45	89.14

13.08%, 2.58%, 10.1%, 16.09%, 33.43%, 3.29%, 5.78%, 19.53%, 35.99%, 8.88%, respectively, and mAP improves 13.69%, indicating that ODDP has better detection ability. Figure 4 shows the tradeoff between “the number of object proposals” and recall, AP respectively. “The number of object proposals” is the hyperparameter of two-stage object detection method. Figure 4 also compare our method with RICNN with fine-tuning [18]. We can see from (a), (b) that as the horizontal axis increases, the recall curve of our method quickly reaches a plateau, while the contrast method requires a larger hyperparameter to reach a plateau, which indicates that our method locate objects more accurate. It can be seen from (c), (d) that our method has higher AP curve in every class and can reach a high AP earlier than the contrast method. The average running time is used to evaluate the speed of each object detection method. As can be seen from Table 1, ODDP runs faster.

4 Conclusion

In this work, a two-stage object detection method for VHR remote sensing image is proposed. Inspired by the dropout method, we first propose Deep Pruning, which can reduce the network capacity and the probability of overfitting and accelerate network

computing. Then, we propose the object detection training algorithm based on Deep Pruning, including LRPN algorithm and OCN algorithm. The region proposal network and classification network can be obtained by inputting the initial network into LRPN and OCN respectively. And the two networks are combined into the two-stage object detection model. The experimental results show that ODDP outperform the existing object detection methods in remote sensing dataset.

Acknowledgements. This work is supported by the National Natural Science Foundation of China (61472161), Science & Technology Development Project of Jilin Province (20180101334JC), Natural Science Foundation of Hunan Province (No. 2018JJ3479).







References

1. Yang, Y., Zhuang, Y., Bi, F., Shi, H., Xie, Y.: M-FCN: effective fully convolutional network-based airplane detection framework. *IEEE Geosci. Remote Sens. Lett.* **14**(8), 1293–1297 (2017)
2. Zhong, Y., Fei, F., Zhang, L.: Large patch convolutional neural networks for the scene classification of high spatial resolution imagery. *J. Appl. Remote Sens.* **10**(2), 025006 (2016)
3. Luo, Q., Shi, Z.: Airplane detection in remote sensing images based on object proposal. In: 2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), pp. 1388–1391. IEEE Press (2016)
4. Liu, K., Mattyus, G.: Fast multiclass vehicle detection on aerial images. *IEEE Geosc. Remote Sens. Lett.* **12**(9), 1938–1942 (2015)
5. Cao, Y., Niu, X., Dou, Y.: Region-based convolutional neural networks for object detection in very high resolution remote sensing images. In: 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), pp. 548–554. IEEE Press (2016)
6. Zhang, R., Yao, J., Zhang, K., Feng, C., Zhang, J.: S-CNN ship detection from high-resolution remote sensing images. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLI-B7, pp. 423–430 (2016)
7. Chen, Z., et al.: Vehicle detection in high-resolution aerial images via sparse representation and superpixels. *IEEE Trans. Geosci. Remote Sens.* **54**(1), 103–116 (2016)
8. Han, J., Zhang, D., Cheng, G., Guo, L., Ren, J.: Object detection in optical remote sensing images based on weakly supervised learning and high-level feature learning. *IEEE Trans. Geosci. Remote Sens.* **53**(6), 3325–3337 (2015)
9. Shao, W., Yang, W., Liu, G., Liu, J.: Car detection from high-resolution aerial imagery using multiple features. In: 2012 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), pp. 4379–4382. IEEE Press (2012)
10. Long, Y., Gong, Y., Xiao, Z., Liu, Q.: Accurate object localization in remote sensing images based on convolutional neural networks. *IEEE Trans. Geosci. Remote Sens.* **55**(5), 2486–2498 (2017)
11. Jiang, Q., Cao, L., Cheng, M., Wang, C., Li, J.: Deep neural networks-based vehicle detection in satellite images. In: 2015 International Symposium on Bioelectronics and Bioinformatics (ISBB), pp. 184–187. IEEE Press (2015)
12. Ševo, I., Avramović, A.: Convolutional neural network based automatic object detection on aerial images. *IEEE Geosci. Remote Sens. Lett.* **13**(5), 740–744 (2016)

13. Wu, H., Zhang, H., Zhang, J., Xu, F.: Typical target detection in satellite images based on convolutional neural networks. In: 2015 IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp. 2956–2961. IEEE Press (2015)
14. Diao, W., Sun, X., Zheng, X., Dou, F., Wang, H., Fu, K.: Efficient saliency-based object detection in remote sensing images using deep belief networks. *IEEE Geosci. Remote Sens. Lett.* **13**(2), 137–141 (2016)
15. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 580–587 (2014)
16. Denil, M., Shakibi, B., Dinh, L., De Freitas, N.: Predicting parameters in deep learning. In: Advances in Neural Information Processing Systems, pp. 2148–2156 (2013)
17. Xia, G.S., et al.: AID: a benchmark data set for performance evaluation of aerial scene classification. *IEEE Trans. Geosci. Remote Sens.* **55**(7), 3965–3981 (2017)
18. Cheng, G., Zhou, P., Han, J.: Learning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images. *IEEE Trans. Geosci. Remote Sens.* **54**(12), 7405–7415 (2016)
19. Cheng, G., Han, J., Zhou, P., Guo, L.: Multi-class geospatial object detection and geographic image classification based on collection of part detectors. *ISPRS J. Photogramm. Remote Sens.* **98**, 119–132 (2014)
20. Krizhevsky, A., Sutskever, I., Hinton, G. E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105 (2012)



W-Shaped Selection for Light Field Super-Resolution

Bing Su¹(✉) , Hao Sheng¹ , Shuo Zhang¹ , Da Yang¹ ,
Nengcheng Chen² , and Wei Ke³ 

¹ State Key Laboratory of Software Development Environment,
School of Computer Science and Engineering, Beihang University, Beijing, China
{su.bing, shenghao, shuo.zhang, da.yang}@buaa.edu.cn

² State Key Laboratory of Information Engineering in Surveying,
Mapping and Remote Sensing (LIESMARS), Wuhan University,
Wuhan 430079, Hubei, People's Republic of China
cnc@whu.edu.cn

³ Macao Polytechnic Institute, Macao, People's Republic of China
wke@ipm.edu.mo

Abstract. Commercial Light-Field cameras provide spatial and angular information, but its limited resolution becomes an important problem in practical use. Different from the conventional images, Light-Field images contain more information of different views that can be used for super-resolution and it makes super-resolution more credible. In this paper, we propose a interpolation based method for Light-Field image super-resolution by taking advantage of the epipolar plane image (EPI) to transfer angular information into spatial information. Firstly, we propose a color recovery framework for undetermined pixels. This framework contains three parts: we estimate the similar-color-diagonal (SCD) for known pixels, we construct a set of filters corresponding to different SCD to generate colors in order to provide a color selection set for undetermined pixel and we propose a W-shaped operator to select a more credible color for undetermined pixel. Finally we use this framework to interpolate EPI and the interpolated EPIs are used to reconstruct a high-resolution image. Experimental results demonstrate that the proposed method outperforms the state-of-art methods for Light-Field spatial super-resolution.

Keywords: Light-field · Super-resolution · Interpolation
W-shaped operator

1 Introduction

The 4D Light-Field imaging introduced by Adelson and Bergen [1] has been considered as the next generation of camera. And with marketization of Light-Field camera, *e.g.* Lytro and Raytrix, it becomes convenient to capture Light-Field

images. A Light-Field camera can acquire both the spatial and angular information of light ray distribution in space. Therefore, it provides more information that reveals the structure of a scene. But due to restricted sensor resolution, its spatial and angular resolution are not high enough for some applications.

To solve this problem, many studies have focused on spatial super-resolution using traditional image super-resolution methods [9, 19]. And many studies focused on video super-resolution processing methods [7, 11]. In recent years, the application of deep learning technology is more and more extensive. Dong *et al.* [5] developed a deep convolutional neural network for traditional image super-resolution. Kim *et al.* [8] proposed a Deeply-Recursive Convolutional Network for traditional image super-resolution. These methods can be used for LF image super-resolution when images of different views are seen as traditional images. Recently, Yoon *et al.* [15] developed a deep convolutional neural network (CNN) to realize LF image super-resolution. They developed the spatial and angular networks independently and fine-tuned via end-to-end training. However, when they realized spatial super-resolution, they did not involve the structure information and they did not consider the influence of occlusions so that the estimation errors increase along occlusion boundaries. Mitra *et al.* [10] proposed a framework for light field denoising and super-resolution. They modeled light field patches using a Gaussian mixture model (GMM) and then reconstructed images based on the estimated disparity. Wanner *et al.* [13, 14] used interpolation lines in EPIs based on each pixel’s direction to realize super-resolution. But their result are not so accurate. Some super-resolution and view synthesis methods are developed based on specifically designed depth maps [4, 18]. But their methods rely on accurate depth estimation [6, 12].

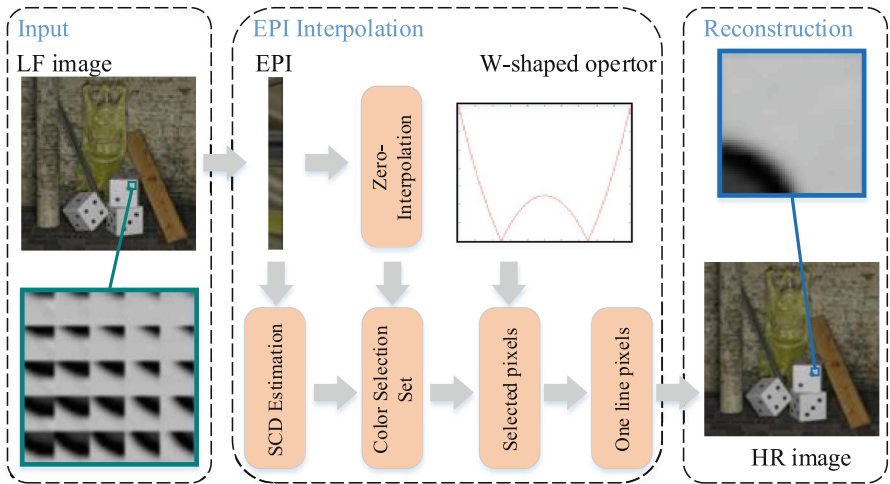


Fig. 1. Our proposed framework is able to realize the transformation of angular information into spatial information and recover high resolution center view image.

Considering the special structure in light field images, we find that the color domain has continuity change in the process of super-resolution and a generated pixel has a more dependent relation on one of the known pixels close to it especially for occlusion regions. Moreover, diagonals generated by stereo-matching based method have correspondence with depth, so the continuity relation also exist between undetermined pixel and the pixels close to it.

We propose that stereo-matching [3] based similar-color-diagonal (SCD) estimation method is more suitable for super-resolution than depth estimation especially for the pixels along occlusion boundaries. SCD not only contains color information but also has a certain correlation with depth. Similar with the construction of filters used in the spinning parallelogram operator (SPO) method [17], a set of interpolation filters corresponding to different SCD are constructed. Therefore we propose an interpolation based framework to realize spatial super-resolution as shown in Fig. 1.

A W-shaped operator is proposed to select a credible color for undetermined pixel. The effect of this selection function makes the transition of boundary pixels in super-resolved image softer. In order to obtain a complete high-resolution image of center view, we also propose a method which interpolates EPI in two stages. After interpolation of one direction’s EPI, we put the interpolated pixel-blocks back to image. Then we extract EPIs along the other direction and interpolate them. Experimental results show that our method achieves better performance than existing methods, both in the occlusion and texture regions.

2 Color Selection via W-Shaped Operator

For a 4D light field $L(x, y, s, t)$, where x and y are the spatial dimensions and s and t are the angular dimensions, one outstanding benefit of LF images over conventional images is access to Epipolar Plane Image (EPI) [2], which is 2D slices of horizontal lines with fixed y^* and fixed t^* , denoted as $E_{y^*, t^*}(x, s)$. There are two advantages of using EPI. Firstly, it contains both the angular and spatial information. Secondly there are special performances for some structure features of scene on EPI. The continuity of color information is contained in certain lines. And the performance of depth information on EPI is a set of diagonal lines.

In this section, we propose to take advantage of the continuity of color information and disparity information on EPI and we construct a model that reflects this continuously changing characteristic in order to correct or supplement the missing information in the scene. The main aim of our research is to find a relation between an undetermined pixel and adjacent known pixels. In order to realize this aim, our model mainly contains three parts: (1) Estimate similar-color-diagonal (SCD) for known pixels. (2) Generate color information corresponding to SCD. (3) Select color information for the undetermined pixel.

Then we redefine the problem. We use a to represent the undetermined pixel’s color, a_l and a_r are the color of known pixels close to undetermined pixel on EPI. Our aim is to find $a_{selected}$ which is a more credible color for undetermined pixel.

2.1 Similar-Color-Diagonal Estimation

On EPI, a diagonal who contains pixels with the most similar color has linear relationship with depth. But occlusion and textureless regions will affect this relationship. Instead of using complete depth estimation method, we pay more attention to color relevance than depth correctness. Our aim is to recover high quality color information for undetermined pixels. So we use the stereo-matching based method on EPI to traverse different diagonals and choose the one who has the most similar color to center view pixel as shown in Eq. (1).

$$d = \arg \min_d \left\{ \left| \sum_{s \in d-Diagonal} w_s a_s - a_{center} \right| \right\}, \quad (1)$$

where w_s is the weight of coordinate s on diagonal labeled d . a_s is the color information of pixels that correspond to the coordinate s . And a_{center} is the color information of center view pixel. As for a diagonal labeled d , the coordinate s of different views on EPI is not always an integer. In this case, there are two products of color and weight for a view. Then the diagonal who corresponds to the most similar color to center view pixel is obtained.

2.2 Color Recovery via SCD

In this section, our aim is to get a color for an undetermined pixel according to a SCD labeled d . We realize this process by filtering, so we construct a set of filters using weights w_s of different coordinate s on filter as shown in Eq. (2).

$$w_s = \delta_d(s) \times \delta_l(s) \times (\text{ceil}(s) - s) + \delta_d(s) \times \delta_r(s) \times (1 - (\text{ceil}(s) - s)), \quad (2)$$

where s is the coordinate on diagonal labeled d , $\text{ceil}(s)$ is the rounded up integer, $\delta_d(s) = 1$ for coordinates on d -diagonal line and $\delta_d(s) = 0$ for coordinates that are not on this line. $\delta_l(s)$ and $\delta_r(s)$ are similar functions to judge whether coordinates are on left (right) side of the diagonal line. And we define $\delta_r(s) = 1$ when s is an integer as an extra. As for the size of filter, it has the same number of row as EPI and it has a width depend on largest disparity of image. There is a one-to-one correspondence between filters and SCD labels.

Then we recover color information for undetermined pixel using Eq. (3)

$$a = g(d) = \frac{F_d \otimes B_{EPI}}{\sum_{s \in d-Diagonal} \omega_s}, \quad (3)$$

where F_d is the filter who correspond diagonal labeled d , B_{EPI} is the pixel block that is centered on undetermined pixel and has the same size of F_d . a is the result color information of undetermined pixel. ω_s is the weight of corresponding coordinate in filters. $F_d \otimes B_{EPI}$ is convolution process, it calculate the sum of products of nonzero weights on filter F_d and corresponding pixels' color on B_{EPI} . After dividing by the sum of weights, we get the final color.

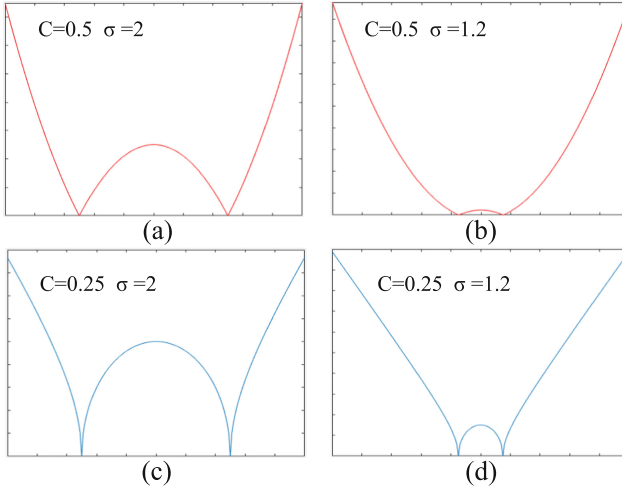


Fig. 2. Functional curve of Eq. (4) when a has only one dimension. The abscissa is color information a , and the ordinate is weight result. For figures in same row, we control the variable c , and for figures in same column, we control the variable σ .

2.3 Undetermined-Pixel Color Selection

After getting SCD for known pixels and the correspondence of SCD and interpolated color, we need to select a more credible color for undetermined pixels. Firstly we generate a selection set $\{d_l, d_l + 1, d_l - 1, d_r, d_r + 1, d_r - 1\}$ for every undetermined pixel. Where d_l is diagonal label of the nearest existing pixel on left side, d_r is diagonal label of the nearest existing pixel on right side. $d_l + 1$ and $d_l - 1$ are diagonal labels close to label d_l . Then according to this SCD set and Eq. (3), a selection set of color information $\{a_{d_l}, a_{d_l+1}, a_{d_l-1}, a_{d_r}, a_{d_r+1}, a_{d_r-1}\}$ is generated. In order to select a more credible color for undetermined pixel, we propose the W-shaped selection function:

$$f_{c,\sigma}(a) = \left| a - \left(a_l + \frac{a_r - a_l}{2\sigma} \right) \right|^c \times \left| a - \left(a_r - \frac{a_r - a_l}{2\sigma} \right) \right|^c, \quad (4)$$

where a_l and a_r are the color informations of known pixels that are close to the undetermined pixel. a is the color information of the undetermined pixel to be decided which corresponds to the choice of diagonal as shown in Eq. (3). Moreover, we can change the value of c and σ to obtain a set of functions. In these equations, it can be seen that this function has two zero points $a_l + \frac{a_r - a_l}{2\sigma}$ and $a_r - \frac{a_r - a_l}{2\sigma}$. We choose one SCD who maximize Eq. (4) to accomplish pixel recovery. The selected a is shown in Eq. (5).

$$a_{selected} = \arg \max_a (f_{c,\sigma}(a)). \quad (5)$$

During the calculation process, we use the rule shown in Eq. (6) to calculate $|a - a_l|$ or $a_l \times a_r$.

$$\begin{aligned} |a - a_l| &= (|a^R - a_l^R|, |a^G - a_l^G|, |a^B - a_l^B|), \\ a_l \times a_r &= \sum_{i=R,G,B} a_l^i \times a_r^i. \end{aligned} \quad (6)$$

Then we analyze the characteristic of this W-shaped selection function. It has two variable parameters in addition to two zeros. Figure 2 shows curve of this function when a has only one dimension. The value of c affects the function curve's rate of change. As for the parameter σ , a larger σ leads to greater weights for the transition zone between zeros. It is more suitable for uniformity-variation zones. On the contrary, a smaller σ is more suitable for occlusion boundaries because a recovered pixel who has color information closer to the known pixels will be given a greater weight.

3 LF Image Super-Resolution

In this section, we use our proposed color selection framework to realize super-resolution. The undetermined pixels in this section are the pixels generated after super-resolution and the known pixels are that before super-resolution. In order to realize our super-resolution processing, we insert zero-lines on EPI in order to reach the same size as high-resolution image. Then with the help of our color selection framework, credible color informations are determined for undetermined pixels. During this process, interpolation filters need to be adjusted in order to fit the change of EPI. And after obtaining interpolated EPIs, a high-resolution image reconstruction framework is proposed.

3.1 Inetrpolation Filter Transformation

After inserting zero-lines in order to get the same size as the image after super-resolution, the rule of transformation need to be found in order to adjust interpolation filters. We stipulate that the center of filter doesn't change. The transformation of the other points is based on the center point, therefore we define the center point as origin of coordinates.

$$s' = \text{ceil}(s) \times \text{upscale} - (\text{ceil}(s) - s), \quad (7)$$

where s is the coordinate who has nonzero value and are related to SCD before transformation and $\text{ceil}(s)$ is the rounded up integer, and upscale is super-resolution multiple. s' is the coordinate after transformation. We use F'_d to represent the filters after transformation.

Then with the transformation of filters, Eq. (3) is also adjusted accordingly as shown in Eq. (8).

$$a = g(d) = \frac{F'_d \otimes B_{EPI}}{\sum_{s' \in d - \text{Diagonal} \cap A_d} \omega_{s'}}, \quad (8)$$

where F'_d is the filter after transformation who correspond diagonal labeled d , B_{EPI} is the pixel block that is centered on undetermined pixel and has the same size of F'_d . a is the result color information of undetermined pixel, and A_d is the set of pixels on EPI who have nonzero color informations. $\omega_{s'}$ is the weight of corresponding coordinate in filters.

3.2 High-Resolution Image Reconstruction

Conventional method for extraction of EPI only contains two orientations. Directly applying interpolation on these two orientations will cause pixel-missing. In order to solve this problem, we extract one orientation’s EPI and realize super-resolution for center view. Then we put pixels of this view back to original LF image. That means it increases the spatial resolution and loses angular resolution in this orientation. But in the other orientation, it still contains spatial information and angular information. In this orientation, novel pixels in a same view can be used to extract EPI and apply interpolation process.

After completion of two stages of interpolation, pixels in the center view are relocated to high-resolution image. Then the high-resolution image reconstruction is accomplished.

4 Experiment

In this section, we evaluate the proposed “selected SCD interpolation” framework on both synthetic and real-world datasets captured by Lytro Illum camera. The quality of the super-resolved image is measured by the peak signal-to-noise ratio (PSNR) and the structural similarity (SSIM) against the ground truth image.

Dataset: The datasets we use in the experiments include the synthetic light field image dataset (HCI) and the real light field image dataset captured by Lytro Illum camera.

We first test the proposed method of our selected SCD based interpolation. After comparing our result with interpolation using the depth ground truth. We find that for most of the occlusion boundaries, using depth ground truth to realize interpolation leads to appearance of blur, because there is diagonal cross on EPI when occlusion occurs. But our framework solved this problem to a certain extent as shown in Fig. 3.

Then we evaluate the effect of our W-shaped operator. We compare the results of interpolation with/without process of SCD selection. Without process of SCD selection means we use one-side SCD for interpolation. The evaluation results are illustrated in Table 1, we observe that compared with the result not using W-shaped operator for SCD selection, our proposed framework shows better quantitative result.

We also compare our result with other methods including methods of light field image super-resolution and conventional image super-resolution. Comparing with these methods, we find that our method shows higher quantitative

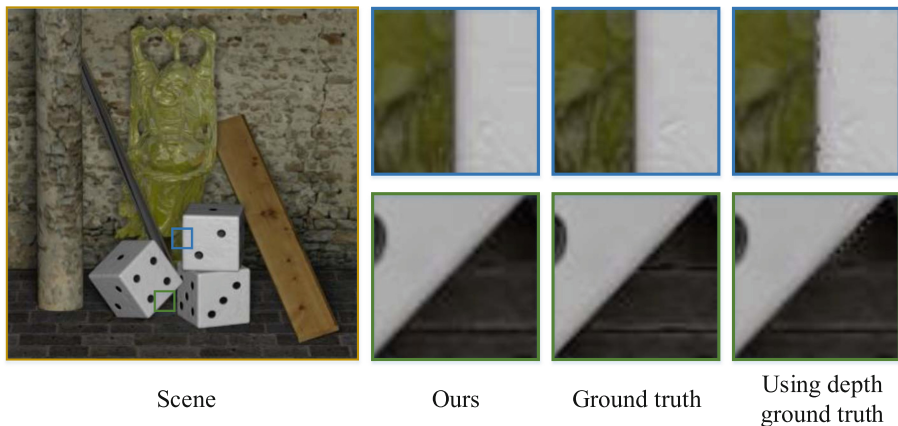


Fig. 3. Comparison of our selected disparity based interpolation and ground truth of depth based interpolation. On the right side, the first column of comparison chart is result of our method. The second column is the ground truth before downsampling and the third column is the result of interpolation using depth ground truth.

Table 1. The evaluation of the disparity selection function. The table value shows the result of $\times 2$ super-resolution.

Data Sets	-Disparity selection		+Disparity selection	
	PSNR	SSIM	PSNR	SSIM
Buddha	39.3104	0.9681	39.4943	0.9684
Mona	38.5082	0.9709	38.8291	0.9709
Papillon	39.8428	0.9759	40.1716	0.9764

Table 2. Quantitative evaluation on the synthetic HCI dataset.

SR Methods	Buddha		Mona	
	PSNR	SSIM	PSNR	SSIM
Mitra [10]	32.37	0.91	34.53	0.95
Wanner [14]	33.83	0.91	36.84	0.94
Zhang [16]	38.27	0.97	34.44	0.97
Yoon [15]	36.94	0.95	37.64	0.96
Ours	39.49	0.97	38.82	0.97

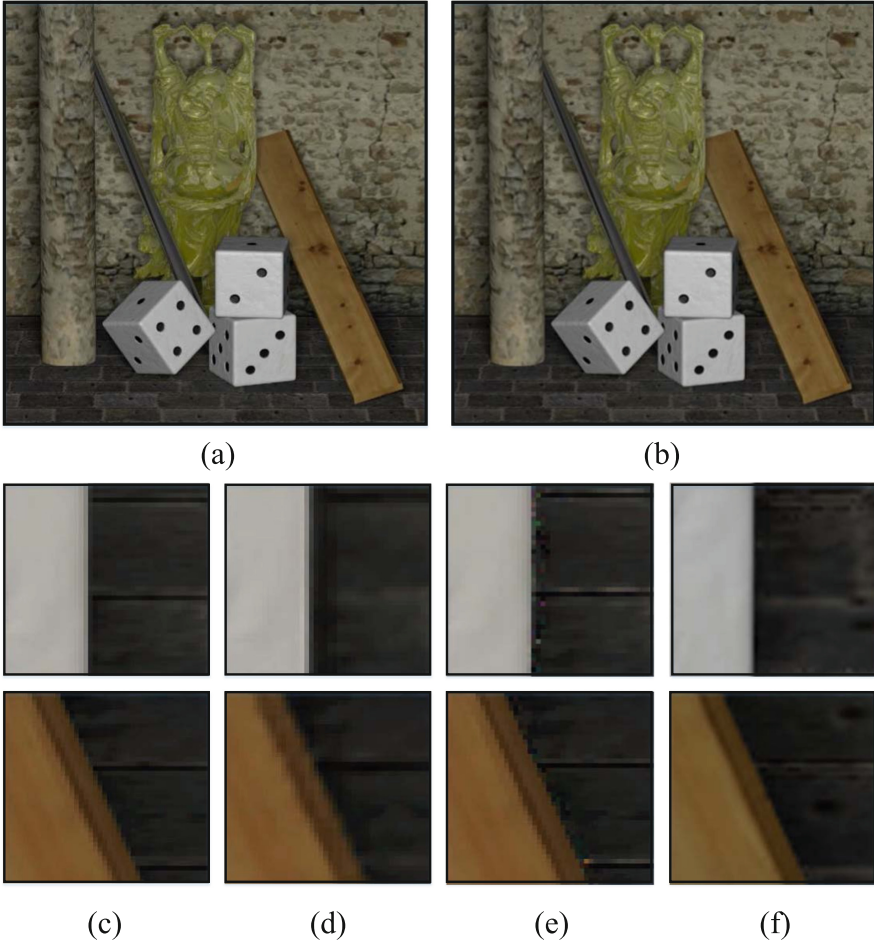


Fig. 4. The super-resolution results using different procedures of the proposed method. (a) Our super-resolved image. (b) Super-resolved result of bicubic-interpolation. (c) Ground truth image. (d) Learning-based reconstruction. (e) Wanner. (f) Our super-resolved image. Our super-resolution method keeps more accurate details especially along depth boundaries.

evaluation results as shown in Table 2. In Fig. 4 we show comparison of some super-resolved image details for light filed image “Buddha”. It can be observed that compared with other methods, our result has clearer borders on occlusion regions. Then we also evaluated our method on the images taken by Lytro camera. Comparing with Zhang *et al.* [16], our results show better appearance especially for occlusion regions as shown in Fig. 5.

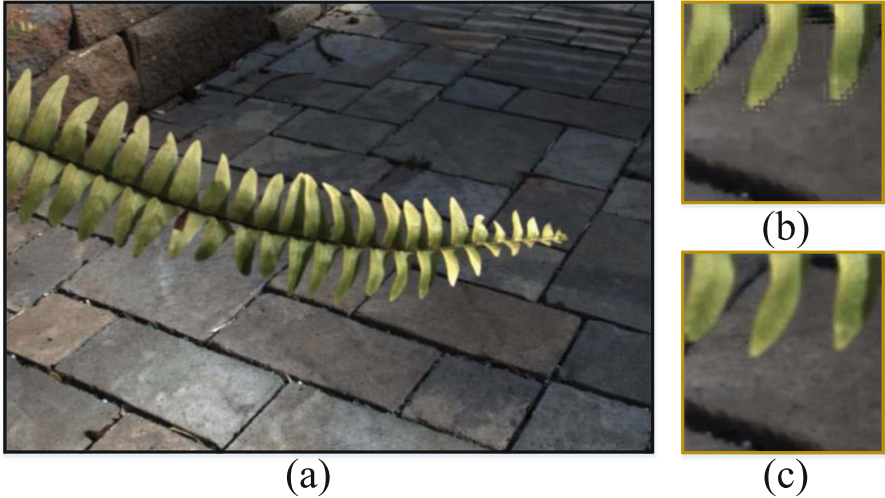


Fig. 5. The super-resolution results of Lytro image. (a) Our super-resolved Lytro image. (b) Result of Zhang *et al.* (c) Ours.

5 Conclusion

Taking into account the special structure of the light field images, we propose a novel interpolation based framework for spatial super-resolution. We also propose a W-shaped operator for SCD selection in order to obtain high-weight color informations for undetermined pixels. We take full advantage of redundant informations that exist in other views to realize a credible retrieval process. During the high-resolution image reconstruction process, we extract and interpolate EPIs in two stages in order to avoid missing pixels caused by direct interpolation. Compared with other super-resolution methods, the proposed method has a better quantitative evaluation result and achieve a better performance on the occlusion regions. Moreover, the undetermined pixel recovery framework is able to be used for other problems like restoration of noise pixels when we define the noise pixels as undetermined pixels.

Acknowledgement. This study is partially supported by the National Key R&D Program of China (No. 2018YFB0505500), the National Natural Science Foundation of China (No. 61635002), the Macao Science and Technology Development Fund (No. 138/2 016/A3), the Program of Introducing Talents of Discipline to Universities and the Open Fund of the State Key Laboratory of Software Development Environment under grant SKLSDE-2017ZX-09, the Project of Experimental Verification of the Basic Commonness and Key Technical Standards of the Industrial Internet network architecture. Thank you for the support from HAWKEYE Group.

References

1. Adelson, E.H., Bergen, J.R.: The plenoptic function and the elements of early vision, pp. 3–20 (1991)
2. Bolles, R.C., Baker, H.H., Marimont, D.H.: Epipolar-plane image analysis: an approach to determining structure from motion. *Int. J. Comput. Vis.* **1**(1), 7–55 (1987)
3. Chen, C., Lin, H., Yu, Z., Kang, S.B., Yu, J.: Light field stereo matching using bilateral statistics of surface cameras (10636919), pp. 1518–1525 (2014)
4. Cho, D., Lee, M., Kim, S., Tai, Y.W.: Modeling the calibration pipeline of the lytro camera for high quality light-field image reconstruction. In: *IEEE International Conference on Computer Vision*, pp. 3280–3287 (2014)
5. Dong, C., Loy, C.C., He, K., Tang, X.: Learning a deep convolutional network for image super-resolution. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) *ECCV 2014*. LNCS, vol. 8692, pp. 184–199. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10593-2_13
6. Honauer, K., Johannsen, O., Kondermann, D., Goldluecke, B.: A dataset and evaluation methodology for depth estimation on 4D light fields. In: Lai, S.-H., Lepetit, V., Nishino, K., Sato, Y. (eds.) *ACCV 2016*. LNCS, vol. 10113, pp. 19–34. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-54187-7_2
7. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) *ECCV 2016*. LNCS, vol. 9906, pp. 694–711. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46475-6_43
8. Kim, J., Lee, J.K., Lee, K.M.: Deeply-recursive convolutional network for image super-resolution. In: *Computer Vision and Pattern Recognition*, pp. 1637–1645 (2016)
9. Ledig, C., et al.: Photo-realistic single image super-resolution using a generative adversarial network. In: *Computer Vision and Pattern Recognition*, pp. 105–114 (2017)
10. Mitra, K., Veeraraghavan, A.: Light field denoising, light field superresolution and stereo camera based refocussing using a GMM light field patch prior, pp. 22–28 (2012)
11. Shi, W., et al.: Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network, pp. 1874–1883 (2016)
12. Tao, M.W., Su, J.C., Wang, T.C., Malik, J., Ramamoorthi, R.: Depth estimation and specular removal for glossy surfaces using point and line consistency with light-field cameras. *IEEE Trans. Pattern Anal. Mach. Intell.* **38**(6), 1155–1169 (2016)
13. Wanner, S., Goldluecke, B.: Spatial and angular variational super-resolution of 4D light fields. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) *ECCV 2012*. LNCS, vol. 7576, pp. 608–621. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33715-4_44
14. Wanner, S., Goldluecke, B.: Variational light field analysis for disparity estimation and super-resolution. *IEEE Trans. Pattern Anal. Mach. Intell.* **36**(3), 606–619 (2014)
15. Yoon, Y., Jeon, H., Yoo, D., Lee, J., Kweon, I.S.: Learning a deep convolutional network for light-field image super-resolution, pp. 57–65 (2015)
16. Zhang, S., Sheng, H., Yang, D., Zhang, J., Xiong, Z.: Micro-lens-based matching for scene recovery in lenslet cameras. *IEEE Trans. Image Process.* **PP**(99), 1 (2017). A Publication of the IEEE Signal Processing Society

17. Zhang, S., Sheng, H., Li, C., Zhang, J., Xiong, Z.: Robust depth estimation for light field via spinning parallelogram operator. *Comput. Vis. Image Underst.* **145**(145), 148–159 (2016)
18. Zhang, Z., Liu, Y., Dai, Q.: Light field from micro-baseline image pair. In: *Computer Vision and Pattern Recognition*, pp. 3800–3809 (2015)
19. Zhou, L.Y., Cai-Xia, S.U., Cao, Y.F.: Image super-resolution via sparse representation. *Comput. Eng. Des.* (2016)



Users Personalized Sketch-Based Image Retrieval Using Deep Transfer Learning

Qiming Huo^(✉), Jingyu Wang, Qi Qi, Haifeng Sun, Ce Ge, and Yu Zhao

State Key Laboratory of Networking and Switching Technology,
Beijing University of Posts and Telecommunications, Beijing 100876, China
314762927@qq.com, {wangjingyu,qiqi,sunhaifeng1,gece,zhaoyu}@ebupt.com

Abstract. Traditionally, sketch-based image retrieval is mostly based on human-defined features for similarity calculation and matching. The retrieval results are generally similar in contour and lack complete semantic information of the image. Simultaneously, due to the inherent ambiguity of hand-drawn images, there is “one-to-many” category mapping relationship between hand-drawn and natural images. To accurately improve the fine-grained retrieval results, we first train a SBIR general model. Based on the two-branch full-shared parameters architecture, we innovatively propose a deep full convolutional neural network structure model, which obtains mean average precision (MAP) 0.64 on the Flickr15K dataset. On the basis of the general model, we combine the user history feedback image with the input hand-drawn image as input, and use the transfer learning idea to finetune the distribution of features in vector space so that the neural network can achieve fine-grained image feature learning. This is the first time that we propose to solve the problem of personalization in the field of sketch retrieval by the idea of transfer learning. After the model migration, we can achieve fine-grained image feature learning to meet the personalized needs of the user’s sketches.

Keywords: Personalized sketch-based image retrieval
Deep full convolutional neural network · Transfer learning
Feature extraction

1 Introduction

The difference in the distribution of sketch and natural image statistics results in completely different image scopes. Using the computer to process hand-drawn image feature and to retrieve related natural pictures require some transformation [8] hand-drawn and natural images to make them in the same image domain. In addition, determining user intent from visual search queries is still a public challenge, especially in sketch-based image retrieval (SBIR). The hand-drawn images are ambiguous, and the same hand-drawing can express the semantics of different things. On the other hand, hand-drawings of the same object, drawn by different users, are also different, and eventually the search results after computer

operations are certainly different. Usually, there is a “one-to-many” relationship between the hand-drawing and categories of natural images. If it is desired that the computer can accurately recognize objects represented by the user-drawn image as humans do, it is necessary to add user relevance feedback information. The feedback allows the user to indicate to the system which of these instances are desired or related, and which are not. Based on feedback, the system modifies its search mechanism and tries to return a more optimal picture set to the user [9]. The feedback here serves as an effective tool for extracting image depth semantic information and doing fine grain image analysis.

The main contributions of this paper are as follows. (1) Based on the natural image cross-image scoping method, extracting the bottom pixel-level edge line information of the natural image, which is input to the improved deep full-convolution neural network simultaneously with the hand-drawn image information. After the training, the mean average precision (MAP) of the model evaluation is greatly improved compared with the traditional image algorithm [5–7, 14] and the deep learning algorithm [1, 2, 12, 13] in recent years; (2) As for the problem of “one-to-many” relationships between hand-drawing and the categories of natural images, we propose a data modeling method based on user feedback using the idea of transfer learning [10]. The way is using the user history feedback to adjust the spatial distribution of the subclass images and input hand-drawn image feature vectors on the basis of a general model. Determine the subspace [17] where the fine-grained natural image and the input hand-drawn image are located in the overall feature space of each category. The migrated model completes the fine-grained image retrieval task and satisfies the user’s personalized requirements to the maximum extent possible.

2 Sketch-Based Image Retrieval General Model

As for the general model of sketch retrieval, our goal is to extract the complete image feature information as much as possible. The more complete the sketch feature is, the more the real content of the sketch can be expressed, and the more accurate the sketch matching is. At the same time, this step is also the basis for the training of personalization model training data. The quality of the common model directly affects the training of the feedback process.

2.1 Image Pre-processing and Feature Extraction

In the natural image contour extraction process, we use the global probability of boundary (gPb) edge detection algorithm and the dual threshold processing method to obtain the binary edge map, retaining the strongest edge information of 25% and removing 25% of the weakest edge information. Then, the canny edge extraction algorithm is used to perform the lag threshold processing. And the pixels connected to the strong edges are left and the isolated edge pixels are removed [2]. The resulting image after filling the remaining blank image to a size of 256×256 is then binary processed. The nature images are filled the blank and binary processed by the same way.

2.2 Establish a Joint Loss Function

The construction of label information is based on the relationship between the hand-drawn image X^S and the contour image X^C provided in the data set. First define the input tag Y , which value is 0 or 1. When the i -th hand-drawn image X_i^S and the contour image X_i^C are in the same category, it is a positive sample, and the triplet $\langle X_i^S, X_i^C, Y_i = 0 \rangle$ is constructed. Conversely, it is a negative sample, construct a triplet $\langle X_i^S, X_i^C, Y_i = 1 \rangle$. When the triplet is input into the neural network, X_i^S and X_i^C are used to calculate V_i^S and V_i^C , where $V_i^S = f_N(X_i^S)$ and $V_i^C = f_N(X_i^C)$, f_N is the neural network forward propagation calculation function. Here is the loss function [4]:

$$\sum_{i=0}^{batch_size} (1 - Y_i) \frac{2}{Q} Ew_i^2 + Y_i \times 2Q \times e^{-\frac{2.77}{Q} Ew_i} . \quad (1)$$

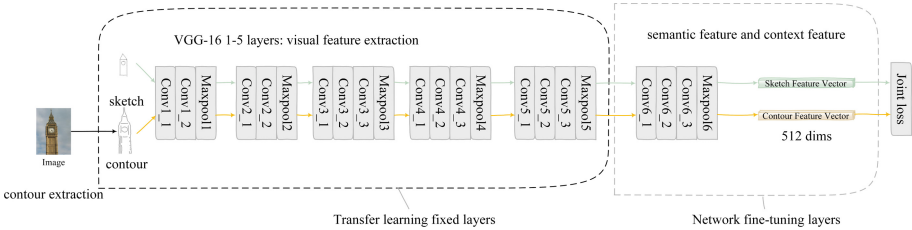


Fig. 1. Two-branch CNN structure and migration learning model outline diagram

Q is a constant, which is the maximum value of $Ew = \|V^S - V^C\|_2$ when the final category is discriminated (Fig. 1).

2.3 Image Similarity Matching and Retrieval

In image retrieval, we calculate the Euclidean distance of the feature vectors of all the pictures in the hand-drawn image and the image library is calculated by traversing the entire list.

$$Sim_{common} = [euc_1, euc_2, euc_3, \dots, euc_n]. \quad (2)$$

Take the index of the top K values as the most similar K pictures as candidate results to return to the user interface. For the R is the set of nature images.

$$index = \arg \min_{i \in R} Euc(i). \quad (3)$$

3 User Feedback Based on Transfer Learning

Through the discussion in the previous section, we can obtain a general model after the training process has converged. When the general model gives candidate results based on the low-level similarity matching, the user is provided with the choice of which pictures are of interest to the user. The system selects the records according to the user history, and judges the positive correlation sample, negative correlation sample, and general correlation sample [16]. Use this data as a training sample for a personalized model. By learning user feedback information, the model will change the distribution of input hand-drawn images and variously related samples in the feature space. Then, based on the rearranged distribution, the system determines the similarity measurement method and retrieves the image with the closest similarity to the user (Fig. 2).

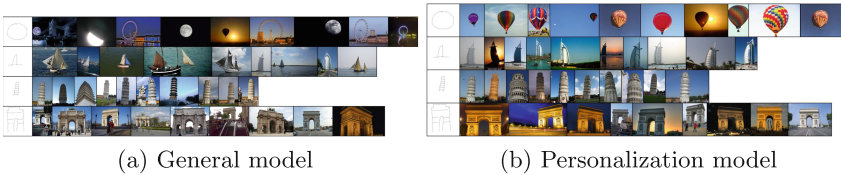


Fig. 2. Different retrieval results of general model and personalization model

3.1 Data Construction

As for the construction of the training sample data set, here we define R is a correlation set of user set U , input hand-drawing set S , user feedback natural image outline image set FI , and construction training data set sampling natural image outline SI set, expressed as $R_{u,s,Fi,Si}$. The set FC represents the parent category of the Fi in the feedback data pair $\langle s, Fi \rangle$, and the set SC represents the sub-category where Fi is located. Defining the input tag $Y \in \{0, 0.5, 1\}$ is used to train the input data of the neural network model. The actual meaning represented is the quantification value of the correlation degree. The rules defining the relationship between samples are defined as follows, in which samples $Si \in SI$. The tag Y rules are defined as follows:

$$Y = \begin{cases} 0, & \text{if } Si \in FC \text{ and } Si \in SC \\ 0.5, & \text{if } Si \in FC \text{ and } Si \notin SC \\ 1, & \text{if } Si \notin FC \text{ and } Si \notin SC \end{cases} . \quad (4)$$

According to the user feedback result data, the training data is constructed with the positive correlation, negative correlation and general correlation sample 1:1:1 ratio when training samples are selected, and the input data format is a quadruple $\langle u, s, Si, Y \rangle$.

3.2 Network Architecture

In this task, since a general correlation is added in this section, the two-branch independent joint loss function (1) based on strong and weak relations in the previous section needs to be rewritten as a three-branch independent function.

$$L_p(Ew, Y) = \delta_1 F_S(Ew) + \delta_2 F_M(Ew) + \delta_3 F_W(Ew). \quad (5)$$

where $Ew = \|V^S - V^C\|_2$ is still the Euclidean Distance between the two output vectors, $F_S(Ew)$, $F_M(Ew)$, $F_W(Ew)$ are the loss functions selected for the positive correlation sample, the general correlation sample, and the negative correlation sample relationship respectively. The prefix term δ is defined here as an independent factor, which is determined according to the value of Y . In order to ensure the independence of branches, set: $\delta_1 = 2 \times |Y - 1| \times (0.5 - Y)$, $\delta_2 = 4 \times |Y - 1| \times Y$, $\delta_3 = 2 \times |Y - 0.5| \times Y$.

As for the loss function of each independent branch, to ensure that the overall feature space does not change, continue to use $F_W(Ew) = 2Q \times e^{-\frac{2.77}{Q}Ew}$ as the branch loss function. For positively correlated samples and generally related samples, since the two parts of loss function are to reduce the distance of output vectors, but magnitude of reduction should different from each other, we introduce a double-threshold method here to control the amplitude. Define: $F_S(Ew) = \frac{2}{Q}(Ew - Th1)^2$ and $F_M(Ew) = \frac{2}{Q}(Ew - Th2)^2$. $Th1$ and $Th2$ are set as thresholds, where $Th1 < Th2 < Q$.

3.3 Similarity Measure and Image Retrieval

The similarity metrics selected during the final test still use the Euclidean distance of the output eigenvector after using the personalized model, so that get a list of similarity results for personalized models:

$$Sim_{personal} = [euc_1, euc_2, euc_3, \dots, euc_n]. \quad (6)$$

Let $w \in [0, 1]$ be the weighting factor, the final similarity list is:

$$Similarity = w \times Sim_{personal} + (1 - w) \times Sim_{common}. \quad (7)$$

Arranged from the smallest to the largest, the indexes of the top K values are the suitable pictures for the user's preference as the final fine-grained search results and returned to the user interface.

4 Experiments and Results

4.1 Dataset and Experiment Settings

The sketch-based image retrieval general model training experiment is based on the public data set Flickr15K. In order to effectively extend the training data and solve the model over-fitting problem during each batch of input data, we

also set up a random hand-drawn image/contour image cropping and flipping to perform data enhancement operations. We use the RMSProp algorithm to train the network for a total of 20 epochs on the Tensorflow platform. For the model to converge to the optimal result, we set the learning rate decay operation, the initial learning rate is set to 0.0001, and the learning rate decay is performed every 5 epochs, and the degree of decay is 0.5. Select 100 between the boundary Euclidean distance Q for the positive and negative sample pairs in the general model (Tables 1 and 2).

As for the personalized model, the feedback images randomly select the natural images in TOP-20 in the general model to simulate the single user selection operation. The sub-category according to the folder name can be determined to have a total of 60. Each experimental set of hand-drawn sketches randomly corresponds to 100 positive correlation samples (positive correlation samples can be repeated). The experiment has also the learning rate decay operation. Using the Adam optimization algorithm to train the network for a total of 20 epochs. Threshold value $Th1 = \frac{Q}{10}, Th2 = \frac{Q}{2}$ is set in the personalized model.

4.2 Model Evaluation

For the general model and personalized model, we can both evaluate the intuitive perception and quantitative values. Intuitively, we can observe the TOP-N results of the search to visually feel the matching of the hand-drawn images with the resulting images. Measure the index of general image retrieval model can use the mean average accuracy (MAP) to determine the precision by category. This indicator can not only show the effect of retrieval on each specific category, but also play a very important role in the generalization of the entire search model (Fig. 3).

Table 1. Neural network structure

Layers	Kernal	Strides	Padding	Filters	Output size
Conv1_1	3 × 3	1	1	64	256 × 256 × 64
Conv1_2	3 × 3	1	1	64	256 × 256 × 64
MaxPool1	2 × 2	2	0		128 × 128 × 64
Conv2_1	3 × 3	1	1	128	128 × 128 × 128
Conv2_2	3 × 3	1	1	128	128 × 128 × 128
MaxPool2	2 × 2	2	0		64 × 64 × 128
Conv3_1	3 × 3	1	1	256	64 × 64 × 256
Conv3_2	3 × 3	1	1	256	64 × 64 × 256
Conv3_3	3 × 3	1	1	256	64 × 64 × 256
MaxPool3	2 × 2	2	0		32 × 32 × 256
Conv4_1	3 × 3	1	1	512	32 × 32 × 512
Conv4_2	3 × 3	1	1	512	32 × 32 × 512
Conv4_3	3 × 3	1	1	512	32 × 32 × 512
MaxPool4	2 × 2	2	0		16 × 16 × 512
Conv5_1	3 × 3	1	1	512	16 × 16 × 512
Conv5_2	3 × 3	1	1	512	16 × 16 × 512
Conv5_3	3 × 3	1	1	512	16 × 16 × 512
MaxPool5	2 × 2	2	0		8 × 8 × 512
Conv6_1	1 × 1	1	0	128	8 × 8 × 128
Conv6_2	3 × 3	1	1	128	8 × 8 × 128
Conv6_3	1 × 1	1	1	512	8 × 8 × 512
MaxPool6	8 × 8	8	0		1 × 1 × 512

Table 2. MAP in general model

Methods	MAP
Ours	0.6449
AFM + QE [15]	0.579
Triplet(fine-tuned final model) [3]	0.3617
Sketchy triplet [13]	0.3591
Query-adaptive re-ranking CNN [1]	0.3230
Triplet loss CNN [2]	0.2445
Siamese CNN [12]	0.1954
PeceptualEdge [11]	0.1837
GF-HOG [6]	0.1222
HOG [5]	0.1093
SIFT [7]	0.0911
SSIM [14]	0.0957

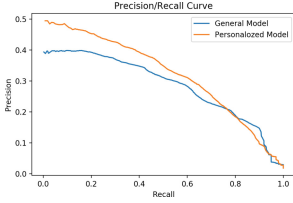


Fig. 3. P-R curves of general and personalized models tested on the entire data set.

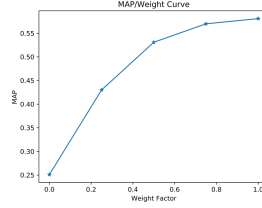


Fig. 4. Personalization Retrieval Model MAP and Weight Factor line chart.

For personalized training model evaluation, in order to make the personalized model contain both sketch content outline information and user feedback preference information, the effect of w is added to the evaluation during the calculation. Select 0, 0.25, 0.5, 0.75 and 1 representative values from $[0, 1]$ as the value of w , and thus the MAP change curve is made according to the value of w as shown in the Fig. 4.

Table 3. Personalization models and general model through fine-grained search sub-categories AP and MAP in Flickr15K comparison tables

Category	1	2	3	5	7	11	12	15	18	30	MAP
General model	0.039	0.012	0.104	0.032	0.44	0.218	0.189	0.784	0.602	0.087	0.2507
Personal model	0.645	0.57	0.417	0.335	0.677	0.393	0.337	0.994	0.654	0.993	0.6015

In the Table 3, it can be seen that while the general model has a high degree of accuracy in retrieving the parent category, it does not achieve good results in the retrieval of natural images in the fine-grained subcategories. The reason is that the sub-category sample features are randomly distributed in the parent category's feature space, so it is probably the correct sample from the same parent category appears, but the sub-category appears randomly. The improved personalized model successfully subdivides the spatial range of the sub-category features. When the input hand-drawn images are calculated to obtain features, according to the principle of distance similarity, the calculated features are located as close as possible to the sub-category samples required by the user, so that sub-category natural images can be efficiently retrieved.

5 Conclusions

In this article, we show how to use an improved depth full convolutional neural network to extract sketch features. We use popular universal datasets to verify the powerful feature extraction capabilities of our designed network. The high-level feature information learned by neural networks is used to improve the

accuracy of sketch retrieval. Based on the idea of transfer learning, the distribution of the migrated feature space is adjusted, and the similarity between the results of user input and historical feedback is further enhanced. The improved personalized model can not only retrieve pictures based on image content but also retrieve pictures based on user selection feedback. However, in the experiment, we used user feedback to enhance supervised sub-tag information to achieve fine-grained sketch retrieval. The accuracy of tagging and user selection operations are highly dependent on the tags. Error tagging or incomplete information of tag information and user's random selection of feedback activities the large probability affects the final retrieval rate. Therefore, in the future, realizing fine-grained sketch retrieval based on weak supervision information needs to design more powerful neural network networks and scientific loss function models.

Acknowledgment. This work was jointly supported by: (1) National Natural Science Foundation of China (No. 61771068, 61671079, 61471063, 61372120, 61421061); (2) Beijing Municipal Natural Science Foundation (No. 4182041, 4152039); (3) the National Basic Research Program of China (No. 2013CB329102).








References

1. Bhattacharjee, S.D., Yuan, J., Hong, W., Ruan, X.: Query adaptive instance search using object sketches. In: Proceedings of the 2016 ACM Conference on Multimedia Conference, MM 2016, pp. 1306–1315 (2016)
2. Bui, T., Ribeiro, L., Ponti, M., Collomosse, J.: Compact descriptors for sketch-based image retrieval using a triplet loss convolutional neural network. *Comput. Vis. Image Underst.* **164**, 27–37 (2017)
3. Bui, T., Ribeiro, L.S.F., Ponti, M., Collomosse, J.P.: Generalisation and sharing in triplet convnets for sketch based visual search. CoRR abs/1611.05301 (2016)
4. Chopra, S., Hadsell, R., LeCun, Y.: Learning a similarity metric discriminatively, with application to face verification. In: CVPR 2005, pp. 539–546 (2005)
5. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR 2005, pp. 886–893 (2005)
6. Hu, R., Collomosse, J.P.: A performance evaluation of gradient field hog descriptor for sketch based image retrieval. *Comput. Vis. Image Underst.* **117**(7), 790–806 (2013)
7. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **60**(2), 91–110 (2004)
8. Ma, Z., Tan, Z., Guo, J.: Feature selection for neutral vector in EEG signal classification. *Neurocomputing* **174**(174), 937–945 (2016)
9. Macarthur, S.D., Brodley, C.E., Kak, A.C., Broderick, L.S.: Interactive content-based image retrieval using relevance feedback. *Comput. Vis. Image Underst.* **88**(2), 55–75 (2002)
10. Pan, S.J., Yang, Q.: A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* **22**(10), 1345–1359 (2010)
11. Qi, Y., et al: Making better use of edges via perceptual grouping. In: CVPR 2015, pp. 1856–1865 (2015)
12. Qi, Y., Song, Y., Zhang, H., Liu, J.: Sketch-based image retrieval via siamese convolutional neural network. In: ICIP 2016, pp. 2460–2464 (2016)

13. Sangkloy, P., Burnell, N., Ham, C., Hays, J.: The sketchy database: learning to retrieve badly drawn bunnies. *ACM Trans. Graph.* **35**(4), 119 (2016)
14. Shechtman, E., Irani, M.: Matching local self-similarities across images and videos. In: *CVPR 2007* (2007)
15. Tolias, G., Chum, O.: Asymmetric feature maps with application to sketch based retrieval. In: *CVPR 2017*, pp. 6185–6193 (2017)
16. Xie, L., Wang, J., Zhang, B., Tian, Q.: Fine-grained image search. *IEEE Trans. Multimed.* **17**(5), 636–647 (2015)
17. Xu, P., et al.: Cross-modal subspace learning for fine-grained sketch-based image retrieval. *Neurocomputing* **278**, 75–86 (2018)



Enhancing Network Flow for Multi-target Tracking with Detection Group Analysis

Chao Li¹ , Kun Qian¹  , Jiahui Chen¹ , Guangtao Xue² ,
Hao Sheng¹ , and Wei Ke³ 

¹ State Key Laboratory of Software Development Environment,
School of Computer Science and Engineering, Beihang University, Beijing, China
{licc,qiankun,chenjh,shenghao}@buaa.edu.cn

² Department of Computer Science and Engineering, Shanghai Jiao Tong University,
Shanghai 200240, People's Republic of China
gt_xue@sjtu.edu.cn

³ Macao Polytechnic Institute, Macao, People's Republic of China
wke@ipm.edu.mo

Abstract. Multi-target tracking (MTT) has been a research hotspot in the field of computer vision. The objective is forming the trajectory of multiple targets in a given video. However, the useful detection and tracklet relationship during the tracking process are not fully explored in most current algorithms and it leads to the accumulation of errors. We introduce a novel Detection Group, which includes the detections within a temporal and spatial threshold and then model the relationship between Detection Group(DG) and close tracklets. Although the minimum-cost network flow algorithm has been proven to be a successful strategy for multi-target tracking, but it still has one main drawback: due to the fact that useful corresponding detection and tracklet relationships are not well modeled, the network flow based tracker can only model low-level detection relationship without high-level detection set information. To cope with this problem, we extend the classical minimum-cost network flow algorithm within the tracking-by-detection paradigm by incorporating additional constraints. In our experiment, we achieved encouraging result on the MOT17 benchmark and our result is comparable to the current state of the art trackers.

Keywords: Network flow · Detection group · Multi-target tracking

1 Introduction

The goal of the multi-target tracking problem is to recover the complete trajectory of a certain number of targets. With the increasing accuracy of pedestrian detectors in recent years, tracking-by-detection paradigm has proven to be effective in crowded and semi-crowded scenes. In this paradigm, tracking can be divided into two separate processes: detection and data association. First, a pedestrian detector is used to acquire the detection set, and then the tracking

result is obtained by correctly assigning these detections to the corresponding trajectories.

The minimum-cost network flow algorithm tends to be unreliable when a relatively long time occlusion or detection error occurred because of its ignorance on the tracklets and other high-level information. To address this problem, we are committed to exploring more intermediate information during the tracking process and incorporating it into the minimum-cost network flow framework. Tracking results using our Detection Group Analysis (DGA) in the enhancing network flow framework are illustrated in Fig. 1.

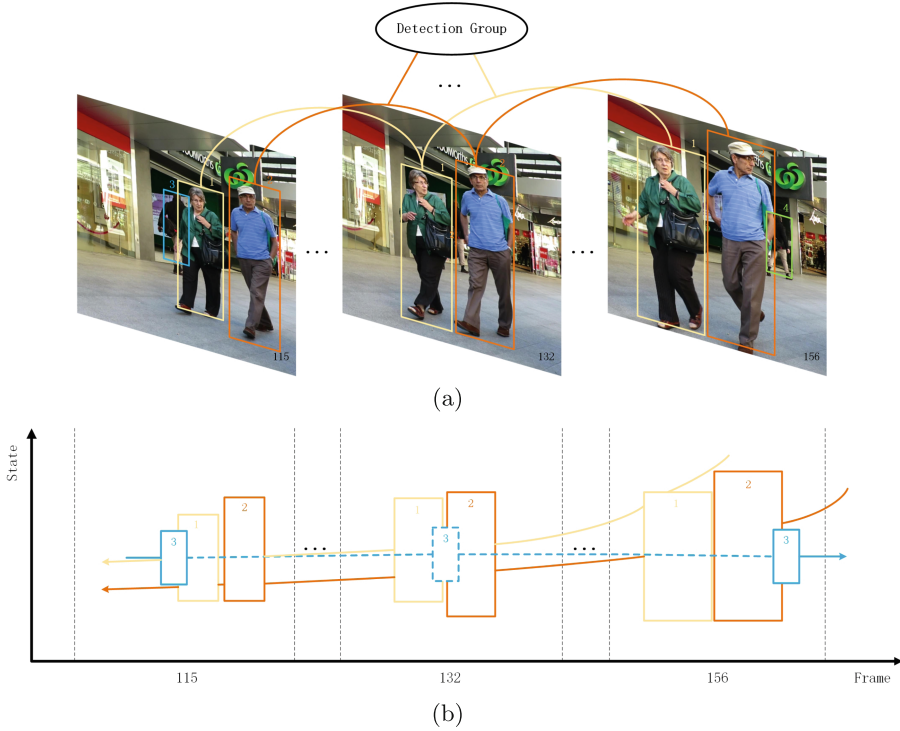


Fig. 1. Results of enhancing network flow tracking with DGA method on MOT17-09-DPM dataset. As the relatively small target (represented by blue and green tracks in (a)) have two identities, because of the occurrence of the two big targets (shown in yellow and orange). This problem is solved by introducing out DGA method, as a complete track (shown in blue in (b)) is formed. (Color figure online)

In summary, this paper makes the following contributions:

- (i) We propose the Detection Group to exploit the relationship of detection sets during the tracking process;
- (ii) We introduce additional constraints in the minimum-cost network flow algorithm to establish the relationship between detection sets;
- (iii) We apply the Detection Group Analysis (DGA) in an iterative way on public dataset and achieve competitive results.

2 Related Work

Most multi-target tracking methods can be divided into two categories [12]: online [2, 6, 11] or offline [5, 7]. Online approach only takes information from past frames into consideration to estimate the current tracking state, which is commonly applied in time-critical scenes.

For the sake of improving the accuracy of multi-target tracking, offline algorithms take information from all frames into consideration. For example, network flow based algorithms [4, 8, 13] define a specific graph, in which each detection are considered as a node, after all the nodes and edges are finely constructed, the tracking problem can be solved quickly by calculating the min-cost network flow on the graph. Conditional random field (CRF) algorithms [9, 10] generalize the global CRF costs to assign label to detections. However, the failure of trajectory association, resulting in trajectory fragmentation and ID-switch or False Negative in above approaches, is still a challenge in above approaches.

3 Proposed Method

3.1 Definition of Detection Group

We define the set consists of temporal and spatial related detections formed by the movement of one or more targets as Detection Group. In more detail, given a set of detections $D = (d_1, \dots, d_n)$, an Detection Group is a collection of several temporally or spatially related detections which is formulated as:

$$U_\rho = (d_{\rho 1}, \dots, d_{\rho k} | \alpha(d_{\rho i}, d_{\rho j}) \leq \delta_t, \beta(d_{\rho i}, d_{\rho j}) \leq \delta_s) \quad \forall 1 \leq i, j \leq k, i \neq j \quad (1)$$

where $\alpha(d_{\rho i}, d_{\rho j})$, $\beta(d_{\rho i}, d_{\rho j})$ are the time intervals and spatial distances of $d_{\rho i}$ and $d_{\rho j}$ respectively, δ_t and δ_s are their corresponding threshold.

3.2 DG-Tracklet Analysis

In this section, we will consider the case when the Detection Group is formed by one or more targets and leads to the fragment of a trajectory and the increase of false negative. In this paper, we find these tracklets quickly and accurately with the help of highly reliable appearance information. Given a set of tracklets $T = (\tau_1, \dots, \tau_m)$, the appearance similarity of tracklet τ_i and τ_j is calculated as:

$$app_{app}(\tau_i, \tau_j) = \omega * (\theta_1, \dots, \theta_\gamma) \quad (2)$$

where

$$\theta_k = \frac{num(app_{det} > \varphi_k)}{num(app_{det} > 0)} \quad (3)$$

$$app_{det}(d_i, d_j) = \frac{app_i \cdot app_j}{\|app_i\|_2 * \|app_j\|_2} \quad (4)$$

$$s.t. \quad \forall 1 \leq i, j \leq k, i \neq j \quad (5)$$

Besides, a novel deep learning appearance feature is used as the appearance model for detection and the similarity between them is evaluated by their cosine value in (4).

Given a tracklet-tracklet set $T = ((\tau_{1s}, \tau_{1e}), \dots, (\tau_{\kappa s}, \tau_{\kappa e}))$ contains κ pairs that are supposed to be connected, for a pair of tracklet (i, j) and interpolation generated detection collection $C = (c_{i+1}, \dots, c_{j-1})$ of them, the penalty item θ is calculated below to represent our confidence in rejecting the detection group hypothesis. We construct dummy node set $S = (s_1, \dots, s_{j-i-1})$, for each node:

$$x_{s_k} = x_{d_i} + \frac{x_{d_j} - x_{d_i}}{j - i} * (k - i) \quad i < k < j. \quad (6)$$

where s_k is the member of C , and x_{s_k} represents the x axis coordinate of it, x_{d_i} and x_{d_j} are the x axis of the last detection in tracklet i and the first detection in tracklet j . Besides, the y axis, width w , height h are all generated in the same way. The occlusion threshold ϕ'_{s_k} is defined as:

$$\phi'_{s_k} = \phi_{d_i} + \frac{\phi_{d_j} - \phi_{d_i}}{j - i} * (k - i) \quad i < k < j. \quad (7)$$

With:

$$\phi_x = \max_{y \in D_{t(x)/x}} \left(\frac{x \cap y}{w_x * h_x} \right). \quad (8)$$

where $t(x)$ are the detections share the same frame with x .

Then, We explore the continuity of points in this sequence that violate the hypothesis. We assume that if a target occurs in the area covered by the Detectin Group within several frames without being fully occluded but not detected by the detector, then we have confidence to reject this hypothesis.

3.3 DG-DG Analysis

Considering the fact that detector is difficult to detect the target behind the Detection Group and discoving all the Detection Groups violently is not a advisable solution, we try to utilise the information of Detection Group formed by a single target within the space-time region to correct some errors during the process of data association. In detail, given two short tracklets i associated with detections (d_{i1}, \dots, d_{im}) and j with (d_{j1}, \dots, d_{jn}) , we construct their dummy nodes separately using Eq. (6) described in Sect. 3.2. Then we compare the degree of overlap between the dummy nodes generated by the two trajectories in the range of t ($\min(m, n)$) frames, and exploiting the coexistence relationship of this pair of tracklets by judging the relationship between their degrees of overlap and correspond thresholds in successive frames. As the overlap function of their detections are defined as:

$$IOU(x, y) = \frac{x \cap y}{x \cup y}. \quad (9)$$

After constructing the sequence of whether the overlap of a specific point higher than the threshold, we have confidence to reject the hypothesis that its overlapped ratio in a period is out of expectation.

3.4 Enhanced Network Flow Approach

In order to establish the relationship between node sets and edges in the network flow algorithm, we define a set of nodes in the network flow framework as a super-node. In this section, we add constraints between edges and super-nodes, super-nodes and super-nodes in the network flow framework to enhance its capability for modeling the tracking problem.

Suppose we have a set of edges $E = (e_1, \dots, e_\delta)$, where e_k in E is associated with detections x_{k1}, \dots, x_{km} , then we treat x_{k1}, \dots, x_{km} as a super-node s_k if we believe that all the elements in E can not coexist with the corresponding S (the set of super-nodes) at the same time, then we can add constraint below in the network flow:

$$s_k = e_k = \frac{x_{k1} + \dots + x_{km}}{m} \quad (10)$$

The equation above means that super-node s_k can be active only if all the nodes that associated with it are 1, its state can only be 0 or 1. Besides, suppose we have noticed that super-node s_i and s_j can not coexist with each other at the same time, then we can add constraint:

$$0 \leq s_i + s_j \leq 1 \quad (11)$$

While s_k represents the existence of a super-node, constraint in (11) guarantees that two super-nodes can not occur at the same time.

Since the super-nodes and factors added by these two constraints are consistent with the constraints of the factors in the classical minimum-cost network flow algorithm, the objective function and solving method of the network flow have not changed. After adding these two constraints, we directly minimize the objective function to get a more accurate solution. In this paper, we use Gurobi [1] to solve integer linear programming.

4 Experiment

We have tested our method on the MOT17 dataset which contains seven training sequences and seven testing sequences, each sequence contains three detection inputs obtained by three different detectors.

Evaluation Metrics. We use CLEAR MOT in [3] to evaluate tracking result. We take the multiple object tracking precision (MOTP \uparrow), multiple object tracking accuracy (MOTA \uparrow), recall (Rcll \uparrow), precision (Prcn \uparrow), the number of mostly ($\geq 80\%$) tracked trajectories (MT \uparrow), the number of partially (20%–80%) tracked trajectories (PT \uparrow), the number of mostly lost trajectories (ML \downarrow), the number of false positives (FP \downarrow), the number of false negatives (FN \downarrow), identity switching (IDs \downarrow) and the number of trajectory fragments (FM \downarrow) into consideration. The symbol \uparrow is a positive indicator that means that the higher the value is, better the result is, while \downarrow means the lower the value is, better the result is.

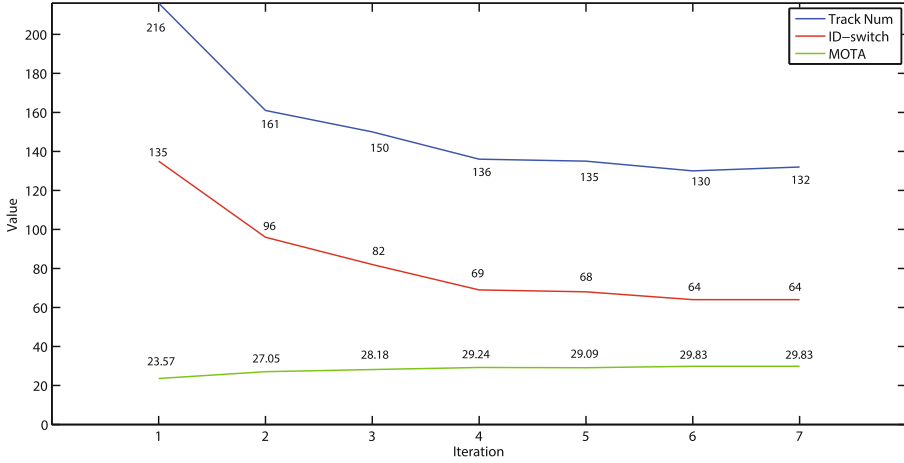


Fig. 2. Our method changes the number of tracks, IDS, and MOTA during the iterative process on the MOT17-02-DPM sequence. With the gradual stabilization of the number of trajectories, MOTA and IDs also tend to be optimal, and the changes tend to be stable.

4.1 Framework Verification

We have implemented a general iterative network flow framework and used its results as the baseline. The baseline result (without using DGA) and final result on MOT17 dataset are shown in Table 1. The specific camera position in every sequences makes MOT17 dataset contains a lot of mutual occlusions between pedestrians.

From the tracking results, we find that the baseline method can generate incomplete track, due to the detector errors caused by the people close to the camera occlude each other, leads to an increased exchange of identities between trajectories. The results prove that our DGA model can handle tracking failure caused by occlusion or interpolation in crowded scenes effectively.

Our DGA framework is based on an iterative network flow approach, We terminate the iteration process when the number of trajectories no longer changes. Experiments have shown that this strategy is effective on most sequences e.g. MOT17-02-DPM. During the process of iteration, the changes of trajectory number, IDs and MOTA on MOT17-02-DPM sequence are presented in Fig. 2). These metrics change in the same way on the other sequences in the MOT17 dataset.

4.2 Evaluation on Public Datasets

To compare with other methods, we tested our method using the same input on the public dataset. Table. 2) shows the quantitative evaluations of our approach and the best previous approaches on MOT17 benchmark. The comparison can also be found in the MOT Challenge website, and our tracker is named IDGA

Table 1. The performance of our method on the MOT17 training dataset

Method	MOTA↑	MOTP↑	Rcll↑	Prcn↑	FP↓	FN↓	IDs↓	FM↓
BaseLine	51.5	84.0	57.1	92.1	16406	144581	2243	2493
Our	52.4	84.1	57.3	92.8	15054	143995	1430	1986

(Iterative Detection Group Analysis). Our tracker achieved competitive results as opposed to the published state-of-the-art trackers.

The work in [7] is the best published method on MOT17. Compared with the Multi Hypothesis Tracking (MHT) based methods [5, 7], they maintain all possible hypotheses and then prune on them, the minimum-cost network flow framework can obtain the global solution more quickly, therefore bring loss of accuracy. However, through our proposed DGA method, the minimum-cost network flow framework can effectively increase the recall rate, track longer trajectories better than the MHT.

Table 2. Results from 2D MOT 2017 Challenge (accessed on 04/27/2018)

Method	MOTA↑	MOTP↑	MT↑	ML↓	FP↓	FN↓	IDs↓	FM↓
PHD_GSDL [6]	48.0	77.2	17.1	35.6	23199	265954	3998	8886
MHT_DAM_17 [7]	50.7	77.5	20.8	36.9	22875	252889	2314	2865
EDMT17 [5]	50.0	77.3	21.6	36.3	32279	247297	2264	3260
EAMTT [11]	42.6	76.0	12.7	42.7	30711	288474	4488	5720
Our	49.9	77.3	22.1	36.7	37060	243148	2426	3846

5 Conclusion

In this paper, we propose the Detection Group to represent the specific kind of detection set, analyse the relationship of Detection Group and tracklet in terms of the introduced DGA model. Furthermore, we embed DGA into the enhancing minimum-cost network flow algorithm to handle long-term occlusion caused by the interaction of targets and track failure caused by the interpolation. The proposed method has shown great performance on the MOT17 dataset, comparable to state of the art performance.

Acknowledgement. This study is partially supported by the National Key R&D Program of China (No. 2017YFC0803700), the National Natural Science Foundation of China (No. 61472019), the Macao Science and Technology Development Fund (No. 138/2016/A3), the Program of Introducing Talents of Discipline to Universities and the Open Fund of the State Key Laboratory of Software Development Environment under grant SKLSDE-2017ZX-09, the Project of Experimental Verification of the Basic Commonness and Key Technical Standards of the Industrial Internet network architecture. Thank you for the support from HAWKEYE Group.

References

1. Gurobi optimization. <http://www.gurobi.com/>
2. Bae, S.H., Yoon, K.J.: Robust online multi-object tracking based on tracklet confidence and online discriminative appearance learning. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1218–1225 (2014)
3. Bernardin, K., Stiefelhagen, R.: Evaluating multiple object tracking performance: the clear mot metrics. *EURASIP J. Image Video Process.* **2008**(1), 246309 (2008)
4. Chari, V., Lacoste-Julien, S., Laptev, I., Sivic, J.: On pairwise costs for network flow multi-object tracking. In: Computer Vision and Pattern Recognition, pp. 5537–5545 (2015)
5. Chen, J., Sheng, H., Zhang, Y., Xiong, Z.: Enhancing detection model for multiple hypothesis tracking. In: Computer Vision and Pattern Recognition Workshops, pp. 2143–2152 (2017)
6. Fu, Z., Feng, P., Angelini, F., Chambers, J., Naqvi, S.M.: Particle PHD filter based multiple human tracking using online group-structured dictionary learning. *IEEE Access* **6**(99), 14764–14778 (2018)
7. Kim, C., Li, F., Ciptadi, A., Rehg, J.M.: Multiple hypothesis tracking revisited. In: ICCV, pp. 4696–4704 (2015)
8. McLaughlin, N., Martinez, Del Rincon, J., Miller, P.: Enhancing linear programming with motion modeling for multi-target tracking. In: Applications of Computer Vision, pp. 71–77 (2016)
9. Milan, A., Leal-Taixé, L., Schindler, K., Reid, I.: Joint tracking and segmentation of multiple targets. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 5397–5406 (2015)
10. Milan, A., Schindler, K., Roth, S.: Detection- and trajectory-level exclusion in multiple object tracking. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 3682–3689 (2013)
11. Sanchez-Matilla, R., Poiesi, F., Cavallaro, A.: Online multi-target tracking with strong and weak detections. In: European Conference on Computer Vision, pp. 84–99 (2016)
12. Shi, X., Ling, H., Xing, J., Hu, W.: Multi-target tracking by rank-1 tensor approximation. In: Computer Vision and Pattern Recognition, pp. 2387–2394 (2013)
13. Zhang, L., Li, Y., Nevatia, R.: Global data association for multi-object tracking using network flows. In: Computer Vision and Pattern Recognition IEEE Conference on 2008 CVPR 2008, pp. 1–8 (2008)



Combine Coarse and Fine Cues: Multi-grained Fusion Network for Video-Based Person Re-identification

Chao Li^{1,2} , Lei Liu¹ , Kai Lv¹ , Hao Sheng^{1,2} , and Wei Ke³ 

¹ State Key Laboratory of Software Development Environment,
School of Computer Science and Engineering, Beihang University, Beijing, China
{licc,leiliu,lvkai,shenghao}@buaa.edu.cn

² Shenzhen Key Laboratory of Data Vitalization, Research Institute in Shenzhen,
Beihang University, Shenzhen, People's Republic of China

³ Macao Polytechnic Institute, Macao, People's Republic of China
wke@ipm.edu.mo

Abstract. Video-based person re-identification aims to precisely match video sequences of pedestrian across non-overlapped cameras. Existing methods deal with this task by encoding each frame and aggregating them along time. In order to increase the discriminative ability of video features, we propose an end-to-end framework called Multi-grained Fusion Network (MGFN) which aims to keep both global and local information by combining frame-level representations with different granularities. The final video features are generated by aggregating multi-grained representations on both spatial and temporal. Experiments indicate our method achieves excellent performance on three widely used datasets named PRID-2011, iLIDS-VID, and MARS. Especially on MARS, MGFN surpass state-of-the-art result by 11.5%.

Keywords: Video-based person re-identification
Multi-grained fusion network · Part-based model
Multi-grained feature

1 Introduction

Person re-identification is a significant task for social security and video surveillance. It aims to retrieve all pedestrians in probe set from a large gallery set in different camera viewpoints. At present, person re-identification is mainly divided into two parts, image-based and video-based person re-identification, and the latter one is closer to realistic scenarios. Both of them should face challenging problems, which include pose variations, complex illumination, multi-camera viewpoints, background clutter and occlusion.

In this paper, we focus on video-based person re-identification, and aim to generate discriminative features from video data. Generating frame-level representations and aggregating them is an intuitive way to encode video data into

discriminative features. There are many models show their effectiveness for generating robust frame-level features [8, 10]. Wei et al. [10] based on pose estimator to extract keypoints, and separates person into three parts. Different from coarsely divide into three parts, they get more precise partitions. Sun et al. [8] proposed a powerful baseline network. They do not directly separate part on original image, instead, they make partition on activation maps. They think the integrating part information can increase the discriminative ability of feature.

After generating features of each frame, a superb aggregation method can promote the performance of original model. Recently, aggregation method for video-based person re-identification is separated into two groups. The first group use temporal pooling to aggregate frame-level features along time. Mclaughlin et al. [6] use Recurrent Neural Network to learn temporal information, and then adopt average pooling to aggregate frame-level features. Another group use weighted average method. There are two ways to generate weights for each frame, attention mechanism and learning by network itself. Zhou et al. [12] adopt RNN hidden state to generate attention map on each frame-level feature. Liu et al. [5] use an independent branch to learn weight of each frame by the network itself. In order to get final video representation, they use temporal average pooling to aggregate weighted frame-level features.

In this paper, simple horizontal partition is adopted to generate fine representation of each person. During this process, fine-grained representations is generated, however, global information is lost. In order to preserve both global and local information, we construct an end-to-end model which combines multi-grained features. In order to preserve both global and local information, we construct an end-to-end model which combines multi-grained features. Totally, our contributions are summarized as two parts:

First, we propose a simple network named Regular Partition Network (RPN). RPN generates representations of each frame which is divided into numbers of partitions firstly, then aggregates them by using temporal pooling along time dimension. Second, we combine multi-grained features of each person, and construct a framework called Multi-grained Fusion Network (MGFN). MGFN combines differently grained features which are generated by three independent branches on spatial, and uses temporal pooling to generate final video features.

2 Proposed Method

Given an input video V contains N frames, $V = \{I_1, I_2, \dots, I_N\}$, and I_n represents the n -th frame in this video sequence. Because ResNet-50 [2] has relatively concise structure and good performance, we use it as a baseline model in this paper. When a video data pass through ResNet-50 before its fully connection layer, the feature of n -th frame is generated as f_n , where $f_n \in \mathbb{R}^D$. After that, we use temporal pooling function TP to aggregate representation of each frame.

$$feat_{baseline} = TP(f_1, f_2, \dots, f_N) \quad (1)$$

$feat_{baseline}$ is a feature of video, $feat_{baseline} \in \mathbb{R}^D$. The subscript of $feat_{baseline}$ shows this feature is extracted by baseline model.

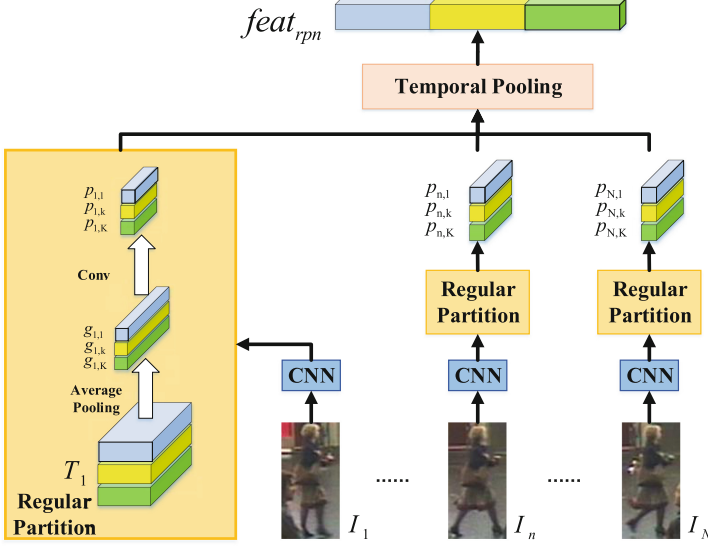


Fig. 1. Regular partition network structure.

2.1 Regular Partition Network

Regular Partition Network. Our regular partition method (Fig. 1) directly makes partitions on activation maps which is generated by ResNet-50. For a better illustration, we compact activation maps extractor and use CNN-block to represent it in Fig. 1. When a video data V pass through RPN, CNN-block is adopted to transform each frame into 3-D tensor and get $T_n \in \mathbb{R}^{H_T \times W_T \times C_T}$ for I_n . Then tensor T_n is separated into K non-overlapped partitions, the size of each partition is $\lfloor H_T/K \rfloor \times W_T$. In next step, T_n is transformed into $g_n \in \mathbb{R}^{K \times C_T}$, where $g_{n,k} \in \mathbb{R}^{1 \times C_T}$ represents the transformed result of $T_{n,k}$ by using average pooling. Especially, the kernel size of average pooling is as same as the size of $T_{n,k}$, where $T_{n,k}$ is the k -th part of T_n . After that, 1×1 2D-convolution is used to reduce the dimension of $g_{n,k}$. Then we get K low dimension vector $p_{n,k} \in \mathbb{R}^{1 \times d}$. Finally, $feat_{rpn} = \theta(P_1, P_2, \dots, P_K)$, where $P_k = TP(p_{1,k}, p_{2,k}, \dots, p_{N,k})$, and θ is concatenation operation.

Training and Testing. During training procedure, we transfer identification task into classification problem. To our empirical practice, the feature of each part should be separated to do classification. The classification loss of P_k is formulated as:

$$loss_k = - \sum_{m=1}^M \log \frac{e^{(W_{k,y_m})^T P_k + b_{k,y_m}}}{\sum_{j=1}^C e^{(W_{k,j})^T P_k + b_{k,j}}} \quad (2)$$

where M is size of a mini-batch in training and C is the class number of classification task. In Eq. 2, $W_k \in \mathbb{R}^{d \times C}$ and $b_k \in \mathbb{R}^C$ is the weights and bias of

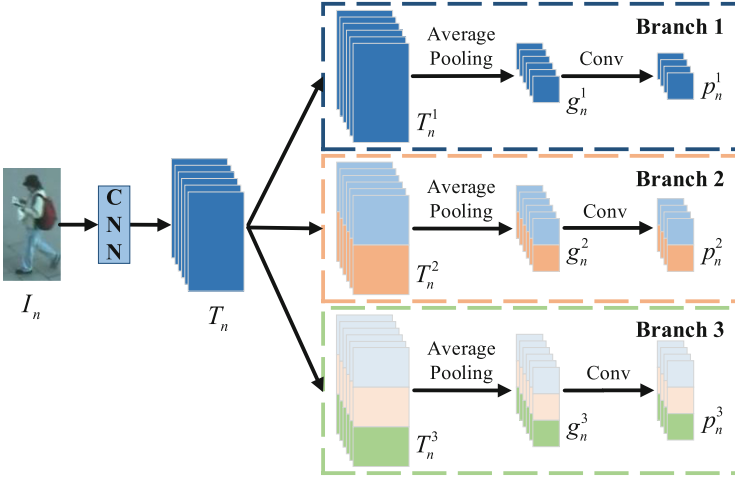


Fig. 2. Multi-grained fusion network structure for single frame.

classifier, and subscript y_m means the ground truth label of i -th sample in this mini-batch. As for the whole RPN model, loss function is defined as:

$$loss_{rpn} = \frac{1}{M \times K} \sum_{k=1}^K loss_k \tag{3}$$

where K is the number of partition. During testing period, $feat_{rpn}$ is used as whole video representation. Using Euclidean distance as evaluation method, the distance between two person is closer, the probability of being same is higher.

2.2 Multi-grained Fusion Network

Information Complementary. In our experiments on Regular Partition Network, we divide middle representation of each frame into horizontal stripes. To our empirical practice the more stripes are divided, the finer features are extracted. Because of the increasing computation and decreasing relevance of data, the stripe should not be too thin. We consider that different partition numbers mean differently grained representation, and combining diversely grained features can keep more information. For achieving our motivation, we construct a framework which fuses global and local cues, and we call it Multi-Grained Fusion Network (MGFN). MGFN combines features with different granularities, and it makes a complementary between local and global information.

Structure of Multi-grained Fusion Network. Multi-grained Fusion Network has multiple branches, and different branch generate a feature with different granularity. In our model, we set branch number to 3 as Fig. 2 shows, and for each independent branch we set the number of partition to 1, 3, 6

(The number of partitions 1, 2, 3 in Fig. 2 is just for better illustration) separately. As Fig. 2 shows, the number of parts K_1 in the top branch is set to 1, and aims to keep global information. $K_2 = 3$ in the middle branch proposes to keep finer-grained information, and $K_3 = 6$ in the bottom branch is expected to keep the finest-grained cues. For easy explanation, we describe the process of MGFN for each frame. Before partition, an input frame I_n is transferred into T_n by using shared CNN-block. Then in i -th branch, T_n is divided into K_i parts, and the partition rules are as same as RPN. We donate k -th part of T_n^i as $T_{n,k}^i$, where $k \in [1, K_i]$. After going through average pooling and 1×1 2D-convolution, p_n^i is generated for i -th branch, where $p_n^i \in \mathbb{R}^{K_i \times C_T}$. Finally, we concatenate p_n^i of each independent branch, and obtain final feature f_n for I_n , where $f_n \in \mathbb{R}^{TK \times d}$, $TK = \sum_{i=1}^3 K_i$ and d is the value of reduced dimension. As for video feature, we also use temporal pool function TP to generate $feat_{mgfn} = TP(f_1, f_2, \dots, f_N)$.

Training and Testing. During training step, we do not combine part feature together as same as the training process of RPN. Firstly, we use temporal pool function TP to aggregate part features in the same location along time. $P_{k_i}^i = TP(P_{1,k_i}^i, P_{2,k_i}^i, \dots, P_{N,k_i}^i)$, where i represent the branch ID, $i \in \{1, 2, 3\}$, k_i is part location in i -th branch, $k_i \in [1, K_i]$. We also regard identification task as classification problem, so fully connection layer is used to change the dimension of $P_{k_i}^i$ to satisfy classification jobs. The $loss_{k_i}^i$ is defined as

$$loss_{k_i}^i = -\frac{1}{M} \sum_{m=1}^M \log \frac{e^{(W_{k_i, y_m}^i)^T P_{k_i}^i + b_{k_i, y_m}^i}}{\sum_{j=1}^C e^{(W_{k_i, j}^i)^T P_{k_i}^i + b_{k_i, j}^i}} \quad (4)$$

where $loss_{k_i}^i$ represent k_i partition loss value of i -th branch, $W_{k_i}^i$ and $b_{k_i}^i$ is weights and bias of classifier for k_i parts in i -th branch. And M is size of mini-batch, C is the class number, y_m is ground truth label of the m -th sample.

$$loss_{mgfn} = \frac{1}{TK} \sum_{i=1}^3 \sum_{k=1}^{K_i} loss_k^i \quad (5)$$

TK is the total number of partition, $TK = \sum_{i=1}^3 K_i$. In testing period, we extract $feat_{mgfn}$ for each person firstly. Then using the same evaluation method and protocol to compute the similarity between identities as RPN.

3 Experiments

3.1 Implementation Details

We evaluate our proposed methods on three widely used video-based person re-identification dataset: PRID-2011 [3], iLIDS-VID [9] and MARS [11]. For PRID-2011 and iLIDS-VID, we use the same evaluation protocol as Wang et al. [6]. As for MARS, we follow the evaluation protocol from Zheng et al. [11]. CMC rank-1 is computed on all the three datasets, mean average precision (mAP) is adopted on MARS at the same time. We sample $N = 16$ consecutive frames as

input from each image sequence, and each adjacent input has 50% overlapped frames for generating more data on training process. Image preprocessing and augmentation are also used to enlarge training set. We first pretrain the baseline model and RPN on DukeMTMC-reID [7] without temporal pooling, then fine-tune them on PRID-2011, iLIDS-VID with temporal pooling. During training process for MGFN, we use pretrained weights on RPN to initialize each branch. For RPN, we set $K = 6$ as Sun et al. [8] has proved in their work. Different from Sun et al. [8], we use 1×1 convolution to reduce the dimension of g_n from 2048 to 256. Except baseline model image size is 256×128 , we resize the frame to 384×128 . Our proposed model uses batched stochastic gradient descent. And we set learning rate to 0.1 at beginning, then drop it to 10% after each 20 epochs.

3.2 Experiments Analyses

Max or Average pooling. In our methods, we utilize temporal pool function to aggregate features of each frame for generating final video representation. Widely used temporal pooling methods include temporal max pooling and temporal average pooling. Temporal max pooling aims to find out the salient value in feature vectors, while temporal average pooling attempts to dilute each value. We compare the performance of these two temporal pooling methods, and the results are summarized in the top group of Table 1. We find that temporal average pooling performs better on both PRID-2011 and iLIDS-VID. One possible explanation is that temporal average pooling method take all information into consideration, so it has a superb ability of anti-interference. So our following experiments use temporal average pooling as default, unless special notification.

Table 1. Comparison of temporal pooling method and classifier parameters shared or not. ResNet: baseline model ResNet-50. MAX: temporal max pooling. AVG: temporal average pooling. Share: classifier parameters are shared. NotShare: classifier parameters are not shared. CMC Rank-1, Rank-5, Rank-10 accuracy (%) are shown.

Models	PRID-2011			iLIDS-VID		
	Rank-1	Rank-5	Rank-10	Rank-1	Rank-5	Rank-10
ResNet+MAX	61.8	81.3	86.7	38.3	63.4	73.7
ResNet+AVG	72.1	94.0	97.8	47.8	75.1	85.5
RPN+Share	86.5	96.7	98.1	66.5	87.5	93.9
RPN+NotShare	88.5	98.1	99.4	69.2	90.1	95.0

Share Parameters or Not Share. In Regular Partition Network, for each partition feature we utilize a classifier to determine this part belongs to which identities. There are two parameters in the classifier, W and b . The intuitive consideration is whether use shared parameters for all classifier. For clearly explanation, we donate the structure of not share parameters as NSP and share parameters as SP . We compare these two structures, and record the performance

in the bottom group of Table 1. The results of experiments show *SP* is more inferior than *NSP*. For different parts, *NSP* uses specialized classifier which more pertinent with each part, however, *SP* only uses a general classifier. In intuition, specialized one must be more superior than general classifier. Even though, *NSP* introduces more parameters, and the increasing computation can be bared.

Table 2. Comparison of our method with state-of-the-art methods. CMC Rank-1 accuracy (%) and mAP (%) are shown. R-1: CMC Rank-1 accuracy (%)

Methods	PRID-2011	iLIDS-VID	MARS	
			R-1	mAP
mvRMLLC+Alignment [1]	66.8	69.1	-	-
CNN+RNN [6]	70.0	58.0	-	-
QAN [5]	90.3	60.8	-	-
Mars [11]	77.3	53.0	68.3	49.3
SeeForest [12]	79.4	55.2	70.6	50.7
end-to-end AMOC+EpicFlow [4]	83.7	68.7	68.3	52.9
ResNet-50	72.1	47.8	-	-
RPN (ours)	88.5	69.2	-	-
MGFN (ours)	90.9	72.8	82.1	56.8

Comparison with State-of-the-Art. Table 2 shows the performance of our methods and other state-of-the-art methods. In Table 2, using ResNet-50 as a feature extractor for each frame is able to get a competitive result. Based on this powerful feature extractor, RPN makes improvements 16.4% and 21.4% on PRID-2011 and iLIDS-VID separately. Future more, MGFN improves the results through combining diversely grained features generated by RPN with the different number of partitions. Especially on MARS, MGFN surpasses the method of Zhou et al. [12] by 11.5%. Zhou et al. proposed a complex model based on six spatial RNNs and temporal attention. In contrast, MGFN is conciser on structure and easier for training. MARS is the most challenging dataset, because of distractor tracklets. Finer performance suggests that our Multi-grained Fusion Network is effective for video-based person re-identification in complex scenarios.

4 Conclusion

In this paper, we propose two methods for video-based person re-identification. One is Regular Partition Network (RPN), the other is Multi-grained Fusion Network (MGFN). RPN adopts partition cues to keep local information. Our experiments indicate RPN shows competitive performance on each video-based dataset. Based on RPN, we construct MGFN to combine differently grained information together, and aim to keep both global and local cues. According to our experiments, MGFN makes remarkable performance although in challenging scenarios.

Acknowledgement. This study is partially supported by the National Key R&D Program of China (No. 2017YFB1002000), the National Natural Science Foundation of China (No. 61472019), the Macao Science and Technology Development Fund (No. 138/2016/A3), the Program of Introducing Talents of Discipline to Universities and the Open Fund of the State Key Laboratory of Software Development Environment under grant SKLSDE-2017ZX-09, the Project of Experimental Verification of the Basic Commonness and Key Technical Standards of the Industrial Internet network architecture. Thank you for the support from HAWKEYE Group.

References

1. Chen, J., Wang, Y., Tang, Y.Y.: Person re-identification by exploiting spatio-temporal cues and multi-view metric learning. *IEEE Sig. Process. Lett.* **23**(7), 998–1002 (2016)
2. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: *Computer Vision and Pattern Recognition*, pp. 770–778 (2016)
3. Hirzer, M., Beleznai, C., Roth, P.M., Bischof, H.: Person re-identification by descriptive and discriminative classification. In: *Scandinavian Conference on Image Analysis*, pp. 91–102 (2011)
4. Liu, H., Jie, Z., Jayashree, K., Qi, M., Jiang, J., Yan, S., Feng, J.: Video-based person re-identification with accumulative motion context. *IEEE Trans. Circ. Syst. Video Technol.* **PP**(99), 1–1 (2017)
5. Liu, Y., Yan, J., Ouyang, W.: Quality aware network for set to set recognition. In: *Computer Vision and Pattern Recognition*, pp. 5790–5799 (2017)
6. McLaughlin, N., Rincon, J.M.D., Miller, P.: Recurrent convolutional network for video-based person re-identification. In: *Computer Vision and Pattern Recognition*, pp. 1325–1334. *IEEE* (2016)
7. Ristani, E., Solera, F., Zou, R., Cucchiara, R., Tomasi, C.: Performance measures and a data set for multi-target, multi-camera tracking. In: *European Conference on Computer Vision Workshop on Benchmarking Multi-target Tracking*, pp. 17–35 (2016)
8. Sun, Y., Zheng, L., Yang, Y., Tian, Q., Wang, S.: Beyond part models: person retrieval with refined part pooling. *arXiv preprint [arXiv:1711.09349](https://arxiv.org/abs/1711.09349)* (2017)
9. Wang, T., Gong, S., Zhu, X., Wang, S.: Person re-identification by video ranking. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) *ECCV 2014*. LNCS, vol. 8692, pp. 688–703. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10593-2_45
10. Wei, L., Zhang, S., Yao, H., Gao, W., Tian, Q.: Glad: Global-local-alignment descriptor for pedestrian retrieval. In: *Proceedings of the 2017 ACM on Multimedia Conference*, pp. 420–428. *ACM* (2017)
11. Zheng, L., Bie, Z., Sun, Y., Wang, J., Su, C., Wang, S., Tian, Q.: MARS: a video benchmark for large-scale person re-identification. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) *ECCV 2016*. LNCS, vol. 9910, pp. 868–884. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46466-4_52
12. Zhou, Z., Huang, Y., Wang, W., Wang, L., Tan, T.: See the forest for the trees: joint spatial and temporal recurrent neural networks for video-based person re-identification. In: *Computer Vision and Pattern Recognition*, pp. 6776–6785 (2017)

Data Processing and Data Mining



Understand and Assess People's Procrastination by Mining Computer Usage Log

Ming He¹, Yan Chen¹, Qi Liu¹, Yong Ge², Enhong Chen¹(✉), Guiquan Liu¹, Lichao Liu³, and Xin Li⁴

¹ University of Science and Technology of China, Hefei, Anhui 230026, China
heming01@foxmail.com, ycwustc@mail.ustc.edu.cn,
{qiliuql, cheneh, gqliu}@ustc.edu.cn

² Nanjing University of Finance and Economic, Nanjing, China
ygestrive@gmail.com

³ IBM (China) Investment Co Limited, Beijing 100193, China
llcliu@cn.ibm.com

⁴ iFlyTek Research, Hefei, China
xinli2@iflytek.com

Abstract. Although the computer and Internet largely improve the convenience of life, they also result in various problems to our work, such as procrastination. Especially, today's easy access to Internet makes procrastination more pervasive for many people. However, how to accurately assess user procrastination is a challenging problem. Traditional approaches are mainly based on questionnaires, where a list of questions are often created by experts and presented to users to answer. But these approaches are often inaccurate, costly and time-consuming, and thus can not work well for a large number of ordinary people. In this paper, to the best of our knowledge, we are the first to propose to understand and assess people's procrastination by mining user's behavioral log on computer. Specifically, as the user's behavior log is time-series, we first propose a simple procrastination identification model based on the Markov Chain to assess user procrastination. While the simple model can not directly depict reasons of user procrastination, we extract some features from computer logs, which successfully bridge the gap between user behaviors on computer and psychological theories. Based on the extracted features, we design a more sophisticated model, which can accurately identify user procrastination and reveal factors that may cause user's procrastination. The revealed factors could be used to further develop programs to mitigate user's procrastination. To validate the effectiveness of our model, we conduct experiments on a real-world dataset and procrastination questionnaires with 115 volunteers. The results are consistent with psychological findings and validate the effectiveness of the proposed model. We believe this work could provide valuable insights for researchers to further exploring procrastination.

1 Introduction

With the rapid growth of Information Technology, computer and Internet have been playing an important role in people's daily life. We use softwares on computer and internet to conduct daily work, and meanwhile we also play games or browser online content a lot. Although the computer and Internet bring our lives great advantages, they also result in various problems to our work. Particularly, the unlimited access to Internet bring us much distraction, e.g., procrastination. Procrastination, which is a general issue of people in modern life, is to voluntarily delay an intended course of action despite expecting to be worse off for the delay [17].

As the expansion of emerging procrastination, it has attracted much researchers' attention on exploring the characteristics and influence of procrastination, especially psychologists and sociologists. To be specific, at first, many researches focus on exploring the reasons and personality of procrastination [10, 17]. Interestingly, there also are some works devised procrastination models to depict the nature of procrastination [12, 14, 21]. Furthermore, many psychologists have provided practical methods to overcome procrastination [2, 8, 18, 19]. While most of researches on procrastination are based on questionnaires by psychologists and sociologists, no work has been done by automatic methods on procrastination. As we know, the questionnaire-based measurement has several drawbacks, e.g., subjectivity and labor intensity. On the contrary, it is very promising to automatically identify user's level of procrastination by analyzing user behaviors on computer, i.e., in a complete data-driven way. Actually, we have analyzed user behaviors on computer, and found that factors of procrastination are correlated with people's usage habits on computer. This careful observation reveals that it is possible to understand and assess people's procrastination by mining computer usage log. However, to achieve this goal, there are several challenges or questions. Specifically, how to effectively bridge the gap between user behaviors on computer and psychology theories? How to automatically assess user procrastination after we have built the aforementioned relation? How to evaluate the effectiveness of the proposed procrastination assessment model?

To address the challenges mentioned above, in this paper, we provide a focused study on assessing user procrastination by mining computer usage log. Along this line, we first make a analysis on user computer logs to observe whether users have different time-series patterns. Specifically, while only computer program records in the log could not provide valuable information to explore user behaviors, to understand user behaviors, we label each recorded computer program with a class, e.g., office or media software. Based on these labeled programs, we can acquire user's behavioral patterns over time, which provides a basis for procrastination exploration. As user behaviors on computer are time-series, we propose a simple procrastination assessment model based on the Markov Chain to evaluate user procrastination. To comprehensively understand the procrastination and explore specified reasons of procrastination, we define and extract some features from the computer usage log based on

psychology theories [2, 12, 17], which successfully bridges the gap between user behaviors on computer and psychology theories. With these extracted features, we devise a sophisticated procrastination assessment model by combining the algorithms of GBDT and CLTree, which can automatically assess user’s procrastination. For precisely evaluating the effectiveness of proposed assessment model, we conduct extensive experiments with a real-world dataset and procrastination questionnaires with 115 volunteers. Experimental results are consistent with psychological findings and clearly demonstrate the accuracy of our proposed sophisticated model.

2 Preliminaries

In this section, we first illustrate the nature of procrastination, and then describe the formulation of the procrastination identification problem.

2.1 Procrastination Illustration

In the field of procrastination, Piers Steel, one of the world premier authority on the science of motivation and procrastination, points out that “*Procrastination is to voluntarily delay an intended course of action despite expecting to be worse off for the delay*” [17]. And according to the study [18] of Piers Steel, many factors of users and the task are correlated with procrastination, such as willpower, postponement, expectation and self-value, sensation seeking and so on. To understand the correlation between these factors and procrastination explicitly, we take willpower as an example:

Willpower: The willpower of users is associated with the degree of procrastination. The stronger willpower users have, the milder procrastinator users will be.

2.2 Problem Statement

As we know, procrastination becomes more pervasive among people resulting from the growing usage of Internet and computer for work. Intuitively, procrastination is correlated with users’ computer behavioral logs, e.g., less working time each day implies higher degree of procrastination. In Fig. 1, we show two

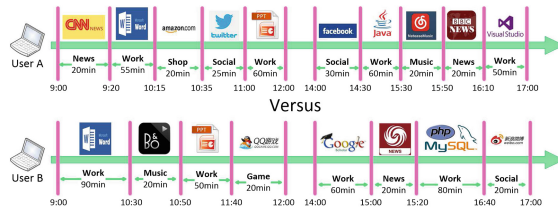


Fig. 1. Examples of two users’ daily sequential behaviors.

users' sequential behaviors on computer. The green axis at the bottom indicates daily time, where the softwares are started and ended based on the associated time. According to Fig. 1, we find that the average working time (56.25 min) of user A is smaller than that of user B (70 min), which shows that user A has lower level of concentration compared with user B. What's more, user A tends to finish tasks in the last minute, while user B tends to finish jobs in advance and is better at managing time. These findings all demonstrate that user A more likely procrastinates on tasks than user B. From this example, we can see that it is possible to assess user procrastination by mining computer usage log.

Formally, given L levels of procrastination degree in which a user should belong to, we wish to capture the accurate level $l_u \in L$ of user u , so the user u can accurately know her procrastination degree and take actions to mitigate the procrastination early. Although we have no prior information on procrastination of users, we can divide users into the different groups in terms of their features F extracted from their computer logs (illustrated in Sect. 4.1) with unsupervised methods. After we obtain each user's group, we can manually label the procrastination level of each group based on the center of the group. And then we acquire every user's procrastination level. Our main assumption in this task is that users with same level of procrastination have similar usage habits on computer.

2.3 Data Collection and Preprocessing

Data Collection. We collected two data sets: users in first dataset are without procrastination levels; users in the second dataset are labeled with procrastination levels.

Unlabeled Dataset. This dataset has 1000 anonymous users' four-week behavioral logs on computer from the Chinese online-data website¹, and contains user behaviors on computer and demographic information:(1) User behaviors: userID, time, procedure name and software name; (2) Demographic information: userID, gender, birthday, education, job, income, province and city. And we filter out the users who have no statistical characteristics and unreasonable records in the data set. As a result, we get 979 users, 10,020 procedures and 7,873,723 records. With this data set, we can build the relation between the behavioral logs and factors of procrastination as described in Sect. 4.1 in details.

Labeled Dataset. Except for the unlabeled dataset, to validate the effectiveness of proposed identification models, we also conduct procrastination questionnaires with 115 volunteers. In psychology, to evaluate user's procrastination, conducting questionnaire on volunteers is a widely used method. For examples, Ferrari et al. [3] introduced that questionnaires are changed according to different tasks (academic procrastination or everyday procrastination). As we want to evaluate user's everyday procrastination and considering the effectiveness and popularity of GPS questionnaire [5], we ask these 115 volunteers to participate in the GPS questionnaire to measure their procrastination level. Specifically, we devise

¹ <http://www.datatang.com/data/43910>.

20 questions that can capture specific reasons (e.g., postponement, willpower) of volunteer’s procrastination. Then, we ask volunteers to choose a score from 1 to 5 for each question. According to the data of returned questionnaires, we can directly label each volunteer’s level (low, middle, high) of procrastination. Considering the validity of returned questionnaires, we collect 46 qualified questionnaires from these 115 volunteers. What’s more, we also record these volunteers’ computer usage log for a month as we have in the unlabeled dataset. Finally, we can use the labeled volunteers to evaluate the accuracy of proposed models for assessing user procrastination.

Softwares Labeling. The original data only has basic information about a procedure (e.g. time, procedure name and software name), without the type of software. Hence, we have crawled type of softwares from a popular software website² and add the type of softwares to the original data. While some softwares can not be labeled by the crawled rules, we label these softwares as “Unknown”.

URLs Labeling. Through analyzing the dataset in depth, 32.31% records on the log are web browsing behaviors and occupy 22.23% computer using time. Intuitively, not all behaviors on browsing webpages are about entertainment, e.g., “ieee.com” probably means a working URL. Therefore, we construct 132 primary rules and 45 secondary rules to label the type (working or entertainment) of URLs. As a result, we can label 79.20% URLs totally, and 20.8% remaining URLs are labeled as “Unknown”.

3 A Simple Model

Inspired by the conclusion of SM Moon et al. in [11] “*Similar users have similar behavioral patterns and contiguous behaviors are correlated with each other*”, we assume that users with similar degree of procrastination have similar behavioral patterns and the previous behavior has an effect on the current behavior. According to this hypothesis, we propose a simple procrastination identification model based on the Markov chain denoted as SPIMMarkovChain.

Formally, for each user u , given her sequential behaviors $\mathbf{B}_u = \{b_{ut}, t \in T\}$, the state-space $\mathbf{SP} = \{sp_s, s \in S\}$, we learn the behavioral types’ transition matrix \mathbf{tp}_u based on \mathbf{B}_u and \mathbf{SP} by the Markov chain. In our application, after labeling all procedures by methods introduced in Sect. 2.3, we can obtain a large number of procedures with types (working, entertainment and unknown). We make the set of procedure types as the state-space $\mathbf{SP} = \{sp_1 : \textit{entertainment}, sp_2 : \textit{working}, sp_3 : \textit{unknown}\}$. As the Markov chain is straightforward, we omitted the detail of learning procedure on the Markov chain. Note that while we focus on the case of a single user u for ease of presentation, the extension to multiple users is easy.

² <http://www.skycn.com>.

By utilizing the Markov model, we acquire all users' transition matrix $\mathbf{TP} = \{tp_u, u \in U\}$ that can approximately represent user's behavior patterns on computer. As mentioned above "Users with similar degree of procrastination have similar behavioral patterns", we naturally cluster users according to the transition matrix. As dimensions of clustering features are not high, there are many clustering methods [1, 16] applied to our clustering problem easily. We choose the K-Means [9] method to cluster users based on the elements of the transition matrix as its simplicity and effectiveness. We choose three elements: $tp(\text{working} | \text{working})$, $tp(\text{working} | \text{entertainment})$, $tp(\text{working} | \text{unknown})$ to cluster users as these elements are inversely proportional to user's procrastination. It is believed that values of above three elements are higher, the procrastination of users is lower.

4 A Sophisticated Model

In this section, we first extract ten features to quantify user procrastination in Sect. 4.1. And then we propose a sophisticated model for procrastination identification, namely Unsupervised Gradient Boosting Decision Tree (UGBDT) model, which can automatically and accurately evaluate user procrastination with extracted features. More details of the UGBDT model are introduced in Sect. 4.2.

4.1 Identification Features

As the SPIMMarkovChain model in Sect. 3 only considers user sequential patterns on computer, it can not depict specified reasons of user procrastination. For example, the SPIMMarkovChain model has identified user A as a serious procrastinator, but it can not provide specified reasons why user A is a serious procrastinator and corresponding solutions what user A should do to mitigate her procrastination. Therefore, we extract ten features from the computer usage log to accurately quantify user procrastination. More importantly, each feature, which comprehensively considers the psychological theories in [2, 12, 17] and the characteristics of users' behaviors on computer, reflects one factor of procrastination. Hence, we devise the following ten features, the first five features of user u are: *Age*, Age_u ; *First Working Time*, fw_t_u ; *Total Working Time*, twt_u ; *Total Entertaining Time*, tet_u ; *Ratio between Work and Entertainment*, rwe_u . The last five are as follows:

Concentration Level: To assess the factor of user's willpower that is a strong index correlating to procrastination based on user's computer usage habit, we measure the average time: $atw_u = \frac{twt_u}{NumWR_u}$ spending on work procedures of user u , where twt_u is defined as above and $NumWR_u$ is the total number of working records in the log of user u . The larger the atw is, the longer concentration u has, and it shows that user has lower degree of procrastination.

Procrastination Length: Intuitively, the more time users spend on playing, the higher procrastination level users have. In terms of this observation, we

devise a feature to the factor of postponement in the strength of procrastination. Given work and entertainment labels of users’ records, we calculate the average playtime apt_u between two adjacent working records of user u .

Daily Behavior Entropy: As lower day regularity and higher disorder reflects lower conscientiousness, we can quantify the factor of user’s conscientiousness by measuring the daily behavior entropy H_u of user u . As proved in psychological theories, higher conscientiousness leads to higher probability on procrastination. For this purpose, we use the Shannon Entropy [7, 15]: $H_u = -\sum_{s=1}^n p_{us} \log p_{us}$ to measure the uncertainty of user’s daily behaviors on computer, where n is the number of softwares user u has used and p_{us} is proportion of using time on software s .

Sensation Seeking: As a user who likes to seek new and adventurous things tends to procrastinate, we measure the sensation seeking of user u by calculating the rate rNT_u between the average using time aut_u of all softwares and the number ns_u of these softwares: $rNT_u = \frac{aut_u}{ns_u}$.

Software Relevance: When a user switches the current software to another one, it partially reflects the user’s procrastination. In terms of this observation, we introduce “Software Relevance” of user u denoted as sr_u to measure the relevance among softwares. Small sr_u means that user u often switches to an irrelevant software from current task, which reflects that the user should spend much more time on going back to previous states and is likely to procrastinate tasks.

As different scales of these features, we first use the transformation $\frac{Max(f)-f}{Max(f)-Min(f)}$ or $\frac{f-Min(f)}{Max(f)-Min(f)}$ to normalize all features into the ranging [0,1], where f is the value of a feature. After having defined and normalized the above features that are covering procrastination’s psychological theories, user’s demographic information and characteristics of computer usage log, we utilize them to identify user procrastination.

4.2 Proposed Model

With these extracted features, we can adopt a clustering method to group users into different levels of procrastination. Although there are many researches about clustering [1, 16], they have some drawbacks for our application. We want to develop a sophisticated model for procrastination identification that can effectively and accurately cluster users based on those extracted features. The gradient boosting decision tree (GBDT) in [4] is an additive regression model utilizing decision trees [13] as the weak learner. And the mode is nicely to our application and has some strengths [20]: (1) Well interpretability by adopting the decision trees over other learners; (2) Less prone to over-fitting by utilizing shallow decision trees. However, the GBDT is a supervised learning method, which can not be applied to our application directly as the datasets have no pre-assigned class labels. Luckily, [6] proposed a novel clustering method called the CLTree based on the supervised learning method decision tree. Inspired by this, we successfully

transform the supervised GBDT classification model to the unsupervised GBDT clustering model via combing the GBDT and the CLTree, which can be used to identify user's procrastination degrees in terms of the extracting psychological features from user behaviors on computer.

Considering details of GBDT, two issues of GBDT should be addressed for clustering items: (1) GBDT can not utilize the decision tree on data without pre-assigned class labels; (2) There are no prior information for optimizing the next decision tree as no training data with labels. Inspired by the CLTree in [6], we nicely solve the two issues by reconstructing the decision tree on data without labels and adopting results of previous decision tree to boost subsequent decision tree.

Issue 1: *Clustering through decision tree* [6].

Liu et al. [6] proposed a CLTree model that is based on a supervised learning technique. Specifically, if the data have several clusters, points are not uniformly distributed in the entire space. Therefore, it is possible to partition the clusters by adding some uniformly distributed points (non-exist points), because within each cluster there are more original points than non-exist points. In terms of this observation, CLTree first regards each data in the dataset with a class Y , and then assumes that the data space is uniformly distributed with non-existing points with label N . By uniformly importing distributed non-existing points on the original data space, the problem of clustering original points turns to classifying original points Y and non-existing points N . In this way, we can adopt the decision tree to solve the transformed classifying problem.

Issue 2: *Unsupervised Gradient Boosting Decision Tree*.

The GBDT learning method utilizes the loss function to gradually boost the effect of next decision tree. Nevertheless, GBDT can not construct a loss function on the data without classes, which results in that GBDT is not capable of improving the effectiveness and accuracy of clusters iteratively. However, considering the solution of issue one, we can use a subset of features space to construct the first unsupervised decision tree by CLTree and obtain preliminary clusters of Y points on the first decision tree. What information can we use from results of the first decision tree? Intuitively, the points in same cluster from the first decision tree have higher probability in same cluster than those points in different clusters. In terms of this observation, we utilize this information to guide next decision tree's construction.

Formally, during current decision tree's construction, if the split makes points within previous decision tree's same cluster into different nodes, we will punish this split as it takes chaos to the node. We take the chaos as a penalty term and is named cluster entropy. By introducing the cluster entropy as a penalty term, we can obtain a adjusted information gain to select the best cut in current decision tree m as following:

$$ag(D, A) = g(D, A) + \sum_{j=1}^J \alpha_j \left(- \sum_{i=1}^n \frac{|D_{Y_{j,i}}^{(m-1)}|}{|D_{Y_j}^{(m-1)}|} \log \frac{|D_{Y_{j,i}}^{(m-1)}|}{|D_{Y_j}^{(m-1)}|} \right), \quad (1)$$

where α_j is the penalty parameter of cluster j , J is the number of clusters from the $(m - 1)$ th tree and $|D_{Y_{j,i}}^{(m-1)}|$ is the number of Y points labeled the $(m - 1)$ th tree’s cluster id j on current decision tree’s node i . By Eq. 1, we can iteratively boost the result of the decision tree by adopting previous cluster results and solve the issue two.

For procrastination identification, we input the extracted features in Sect. 4.1 to the unsupervised GBDT model. On the above, we have successfully transformed the supervised GBDT classifying method to the unsupervised clustering method by combining the CLTree and reconstructing the GBDT. The supervised GBDT also can be used to other classification problems with good interpretability.

5 Experiments

In this section, we conduct extensive experiments on two real-world data sets for validating the effectiveness of proposed models.

5.1 Baseline Methods and Evaluation Metrics

Since user’s procrastination identification is a clustering problem, we adopt KMeans as one baseline, which is a representative clustering method and widely used in practical works. What’s more, our sophisticated UGBDT method refers to the design of CLTree. Spontaneously, CLTree is selected as a baseline, too.

To evaluate the proposed models comprehensively, we adopt two categories of evaluation metrics:

Evaluation Metrics for Psychology. As the purpose of proposed models is to assess user’s level of procrastination, we must evaluate that whether the results conform to the psychological findings on procrastination. Inspired by “The procrastination is correlated with jobs and ages: the procrastination of students is the most serious and the procrastination of users tends to decrease with age” in [18], we extract two psychological findings: (1) The procrastination of students is the most serious compared with other jobs; (2) The procrastination of users tends to decrease with age. We adopt these two findings to validate whether the results assessed by proposed models conform to psychological theories.

Accuracy. As we have one dataset with labeled procrastination levels of people, we intuitively compute the accuracy of proposed models based on the labeled dataset, which can accurately evaluate proposed models on assessing people’s procrastination. Particularly, larger accuracy indicates better performance of procrastination assessment.

5.2 Performance on Unlabeled Dataset

In this section, we present experimental results on unlabeled dataset introduced in Sect. 2 for validating the proposed models’ performance on assessing people’s

procrastination. As people on the first dataset are unlabeled, it means that we do not know the actual level of people’s procrastination. Although the procrastination of these unlabeled users are unknown, they have demographic information (e.g., jobs and ages) that can be utilized to validate the psychological findings on procrastination:

Distributions of Users on Jobs. To validate the finding “The procrastination of students is the most serious compared with other jobs”, we exhibit users’ distributions with different procrastination levels on jobs as shown in Fig. 2. According to Fig. 2, we can draw several implications: (1) The result of UGBDT shows that the procrastination of students (30.6%) is most serious than other jobs, while the job with most serious procrastination of KMeans, MarkovChain and CLTree are “Clerk on government”, “Freelancer” and “Others”, respectively; (2) The result of UGBDT also shows that the procrastination of leaders on government and company is lower than clerk, while other baselines can not validate this psychological finding.

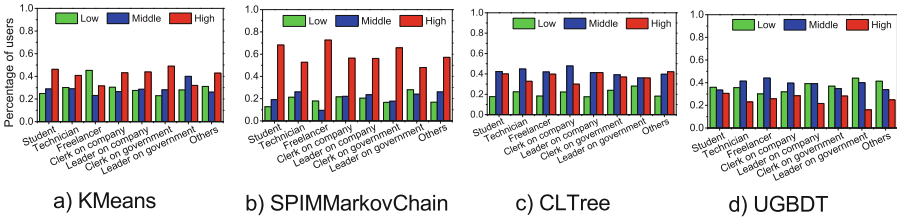


Fig. 2. Distributions of users on jobs with models.

Distributions of Users on Ages. Next, in order to validate the finding “The procrastination of users tends to decrease with age”, we exhibit users’ distributions with different procrastination levels on ages in Fig. 3. According to Fig. 3, we can draw several implications: (1) The results of UGBDT and SPIMMarkovChain show that the degree of procrastination obviously decreases as the age of users increases, while the trend of decrease based on KMeans and CLTree are not obvious compared with UGBDT and SPIMMarkovChain; (2) The users’ distributions of different levels of procrastination by UGBDT and CLTree are more truthful than other two baselines, because users with high procrastination grouping by SPIMMarkovChain and KMeans are much more than users with middle or low procrastination. In terms of these two findings, we find that only UGBDT can both effectively capture the trend of users’ procrastination on ages and actually group users into different levels of procrastination.

5.3 Performance on Labeled Dataset

Except for the experiments on unlabeled dataset, we also conducted experiments on labeled dataset to evaluate the effectiveness of proposed models. Similar to

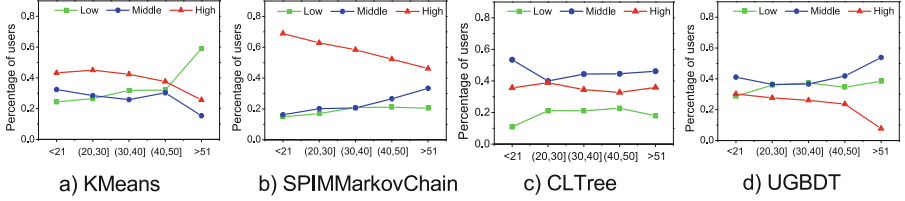


Fig. 3. Distributions of users on ages with models.

Sect. 5.2, we also adopt KMeans and CLTree as baselines to compare the accuracy of proposed models (SPIMMarkovChain and UGBDT). For users on labeled dataset, we can group them into three clusters by these methods, and manually label the three clusters with corresponding procrastination levels by clustering features. Then, we compare each user’s procrastination level by clustering methods with the labeled levels on the dataset. Finally, we can obtain each method’s accuracy on evaluating user’s procrastination level as: *KMeans*, 52.2%; *SPIMMarkovChain*, 43.5%; *CLTree*, 50.0%; *UGBDT*, 60.9%. These results demonstrate that the UGBDT has the largest accuracy (60.9%) compared with other methods, which means that UGBDT is the best method on identifying user’s procrastination. Particularly, the result (52.2%) of KMeans based on the extracted features outperforms better than SPIMMarkovChain (43.5%) based on behavioral patterns, which demonstrates the effectiveness of extracted features.

5.4 Exploration of Procrastination Reasons

Compared with the simple model in Sect. 3, the sophisticated UGBDT model can reveal factors that may cause user’s procrastination based on extracted features. In this section, we show a case study on three questionnaire users with different procrastination level to evaluate the capacity on depicting specified reasons of user procrastination.

We show the captured procrastination reasons of these three users by the sophisticated model. We exhibit the extracted features of these three selected users in Fig. 4. According to Fig. 4, we can find that most of extracted features

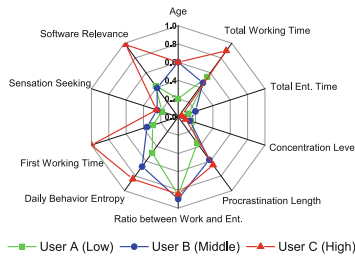


Fig. 4. A case study on three users.

(e.g., “First Working Time” and “Procrastination Length”) of the user with higher procrastination are generally greater than the user with lower procrastination, and that the reasons causing the severe procrastination of user C are severe postponement and lower expectation and self-value, which is conforming to the observation of user questionnaires. If user C wants to mitigate her procrastination, she should improve the action and power of focus. Considering computer usage behaviors, user C should improve the working time and focus on current tasks, avoiding disturbs of other things.

6 The Conclusion

In this paper, we provided a focused study on understanding and assessing people’s procrastination by mining users’ behavioral log on computer. As user’s behaviors on computer are time-series, we first proposed a simple procrastination assessment model based on the Markov Chain, which could directly capture user’s behavioral patterns and evaluate procrastination. However, this simple model could not depict reasons of procrastination. To explore possible reasons of user procrastination, we then extracted some features from computer usage log to quantify procrastination, which successfully bridges the gap between user behaviors on computer and psychological theories. With extracted features, we devised a sophisticated procrastination assessment model by combining the algorithms of GBDT and CLTree, which can accurately assess the level of user procrastination. More importantly, the sophisticated model can reveal factors that may cause people’s procrastination. To validate the effectiveness of proposed models, we conducted extensive experiments with unlabeled dataset and procrastination questionnaires with 115 volunteers, and the experimental results clearly demonstrate the effectiveness of our proposed sophisticated model. To the best of our knowledge, this work is the first to automatically assess people’s procrastination based on computer usage log and successfully transform the supervised GBDT classification model to the unsupervised GBDT clustering model. Also, this work could provide valuable insights for psychology/behavior researchers to further explore procrastination.

Acknowledgements. This research was partially supported by grants from the National Natural Science Foundation of China (Grants No. 61727809, U1605251, 61672483, 61602234 and 61572032), and the Science Foundation of Ministry of Education of China & China Mobile (No. MCM20170507).

References

1. Berkhin, P.: A survey of clustering data mining techniques. In: Kogan, J., Nicholas, C., Teboule, M. (eds.) *Grouping Multidimensional Data*, pp. 25–71. Springer, Heidelberg (2006). https://doi.org/10.1007/3-540-28349-8_2
2. Burka, J.B., Yuen, L.M.: *Procrastination: Why You Do it, What to Do About it Now*. Da Capo Press, Cambridge (2008)

3. Ferrari, J.R., Johnson, J.L., McCown, W.G.: Assessment of academic and everyday procrastination. Procrastination and Task Avoidance. The Springer Series in Social Clinical Psychology, pp. 47–70. Springer, Boston (1995). https://doi.org/10.1007/978-1-4899-0227-6_3
4. Friedman, J.H.: Stochastic gradient boosting. *Comput. Stat. Data Anal.* **38**(4), 367–378 (2002)
5. Lay, C.H.: At last, my research article on procrastination. *J. Res. pers.* **20**(4), 474–495 (1986)
6. Liu, B., Xia, Y., Yu, P.S.: Clustering through decision tree construction. In: Proceedings of the Ninth International Conference on Information and Knowledge Management, pp. 20–29. ACM (2000)
7. Liu, Q., Chen, E., Xiong, H., Ge, Y., Li, Z., Wu, X.: A cocktail approach for travel package recommendation. *Knowl. Data Eng. IEEE Trans.* **26**(2), 278–293 (2014)
8. Liu, Q., Zeng, X., Zhu, H., Chen, E., Xiong, H., Xie, X., et al.: Mining indecisiveness in customer behaviors. In: IEEE International Conference on 2015 Data Mining (ICDM), pp. 281–290. IEEE (2015)
9. MacQueen, J., et al.: Some methods for classification and analysis of multivariate observations. In: Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Oakland, vol. 1, pp. 281–297 (1967)
10. McCown, W., Johnson, J., Petzel, T.: Procrastination, a principal components analysis. *Pers. Individ. Differ.* **10**(2), 197–202 (1989)
11. Moon, S.M., Illingworth, A.J.: Exploring the dynamic nature of procrastination: a latent growth curve analysis of academic procrastination. *Pers. Individ. Differ.* **38**(2), 297–309 (2005)
12. Procee, R., Kamphorst, B.A., Wissen, A.v., Meyer, J.J.C.: An agent-based model of procrastination. In: ECAI 2014–21st European Conference on Artificial Intelligence, 18–22 August 2014, Prague, Czech Republic-Including Prestigious Applications of Intelligent Systems (PAIS 2014), vol. 263, pp. 747–752. IOS Press (2014)
13. Quinlan, J.R.: Simplifying decision trees. *Int. J. Man Mach. Stud.* **27**(3), 221–234 (1987)
14. Ross, D.: Economic models of procrastination. *Legal Ethics* (2010)
15. Shannon, C.E.: A mathematical theory of communication. *ACM SIGMOBILE Mob. Comput. Commun. Rev.* **5**(1), 3–55 (2001)
16. Sim, K., Gopalkrishnan, V., Zimek, A., Cong, G.: A survey on enhanced subspace clustering. *Data Min. Knowl. Discov.* **26**(2), 332–397 (2013)
17. Steel, P.: The nature of procrastination: a meta-analytic and theoretical review of quintessential self-regulatory failure. *Psychol. Bull.* **133**(1), 65 (2007)
18. Steel, P.: The procrastination equation: how to stop putting things off and start getting stuff done. Random House Canada (2010)
19. Wu, L., Liu, Q., Chen, E., Xie, X., Tan, C.: Product adoption rate prediction: a multi-factor view. In: Proceedings of the 2015 SIAM International Conference on Data Mining, pp. 154–162. SIAM (2015)
20. Ye, J., Chow, J.H., Chen, J., Zheng, Z.: Stochastic gradient boosted distributed decision trees. In: Proceedings of the 18th ACM Conference on Information and Knowledge Management, pp. 2061–2064. ACM (2009)
21. Zhu, Y., Zhu, H., Liu, Q., Chen, E., Li, H., Zhao, H.: Exploring the procrastination of college students: a data-driven behavioral perspective. In: Navathe, S., Wu, W., Shekhar, S., Du, X., Wang, X., Xiong, H. (eds.) International Conference on Database Systems for Advanced Applications, vol. 9642, pp. 258–273. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-319-32025-0_17



Group Outlying Aspects Mining

Shaoni Wang¹ , Haiyang Xia¹ , Gang Li² , and Jianlong Tan³

¹ School of Computer Science, Xi'an Shiyou University, Shaanxi 710065, China
snwangxyz@gmail.com, haiyangxia15@gmail.com

² Deakin University, Geelong, Australia
gang.li@deakin.edu.au

³ Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
tanjianlong@iie.ac.cn

Abstract. Existing works on outlying aspects mining have been focused on detecting the outlying aspects of a single query object, rather than the outlying aspects of a group of objects. While in many application scenarios, methods that can effectively mine the outlying aspects of a query group are needed. To fill this research gap, this paper extends the outlying aspects mining to the group level, and formalizes the problem of *group outlying aspect mining*. The Earth Move Distance based algorithm GOAM is then proposed to automatically identify the outlying aspects of the query group. The experiment result shows the capability of the proposed algorithm in identifying the group outlying aspects effectively.

Keywords: Contrast mining · Subspace selection
Group outlying aspects mining

1 Introduction

Many real world big data applications call for one important function of identifying the set of features on which the user interested object is the most distinguished from others. Usually, this object is termed as the query object, and the set of features is termed as the *subspaces* or *aspects*. Accordingly, this research problem is referred to as *outlying subspaces detection* [14] or *outlying aspects mining* [4].

It is worth noting that, *outlying aspects mining* is different from *outlier detection*. In the context of *outlying aspects mining*, the aim is to identify a subset of attributes (aspects or subspace) which makes the query object the most outlying, rather than verifying whether the query object is an outlier or not. What we are interested in the task of *outlying aspects mining* is to explain what aspects make the query object the most different from the rest. In contrast, *outlier detection* [13] aims to identify all possible outliers in the dataset, without explaining why or how they are different. Hence, the outlying aspects mining is also referred to *outlier interpretation* [2] or *object explanation* [8].

Outlying aspects mining has many practical applications, such as examining the symptoms that make a cancer patient most different from other patients, or

identifying the factors that make a consumer most dissatisfied with his deal, or finding out the most distinguish features for a candidate applicant, and so on. However, existing works on outlying aspects mining all focus on detecting the outlying aspects of a single query object, rather than the outlying aspects of a group of objects. While in many application scenarios, practitioners are more interested in the outlying aspects of a particular group instead of a single object. For example, an NBA coach may be interested in what are the competitive advantages or disadvantages of his team when comparing with other teams. A car buyer will be interested in features that differentiate a particular brand/model of cars from the others. A teacher may wonder what is the significant difference between groups of students from consecutive years.

To filling this research gap, this paper aims to identify the outlying aspects of a query group. For this *group outlying aspects mining* problem, there are three research challenges: the first challenge is how to represent the group based on the features of the individuals in the group. Some may consider it trivial as we may simply adopt the arithmetic mean of the features for all individuals. Although theoretically the arithmetic mean of all elements in each feature can describe the features of the group. However, it can easily be affected by outlier values, and fails to provide the overall distribution of values for the group features. The second challenge is how to evaluate the outlying degree of the query group in different aspects (subspaces), that means an appropriate evaluation measure for the outlying degree of a query group on different candidate subspaces is needed. The last challenge we need to tackle is how to improve the efficiency, because when the dimension of the data is high, the candidate subspace which grows exponentially will easily go beyond the limits of the computation resources, so an appropriate pruning strategy is needed to alleviate the curse of dimensionality.

The main contributions of this work to advancing the outlying aspect mining filed are follows.

- We extend the task of outlying aspects mining to the group level and formalize the research problem of *group outlying aspects mining*.
- A novel algorithm named GOAM is proposed to solve the *group outlying aspects mining* problem.

Instead of using the arithmetic means, GOAM algorithm adopts the histogram to represent the group, based on the individuals in the group. Compare with arithmetic means, the histogram representation is not sensitive to outliers, and can capture the group features more accurately. After this representation, the GOAM algorithm utilizes the *Earth Mover's distance* to measure the outlying degree of different groups on candidate subspaces, and a sophisticated pruning technique is designed to tackle the exponentially large search space. Finally the experiment results demonstrate the effectiveness and efficiency of our proposed GOAM algorithm.

Having introduced the motivation and rationale of this research, the rest of this paper is organized as follows. Section 2 reviews of the related works. Section 3 formalizes the problem of *group outlying aspects mining*. Section 4 presents the

GOAM algorithm for mining the group outlying aspects. Section 5 presents the details of our experiment. Section 6 concludes the paper with some suggestions of future work.

2 Related Work

The problem of *outlying aspects mining* is originated from outlier detection. Initially, almost all outlier detection research focused on detecting the objects that are significant different from others. Accordingly, many research work have been devoted to designing measurements for the difference between two objects, such as [1, 5]. Only a few research has been focused on providing the explanations in the outliers detection process, and this line of research is know as the *outlier explanation*. For example, [8] proposed the use of High Contrast Subspace (HiCS) and CMI to identify the subspaces where most of the outliers exist. [9] focus on identifying subspaces where the outlier is separated from its neighbors. [1] focused on detecting outliers in the low dimensional projection subspace of the data. [5] used the subspace outlier degree to identify the outliers in axis-parallel subspaces. [3] focus on identifying the optimal subspace on which an outlier is most deviated from its neighbors. However, although these methods offer the explanations by identifying outliers in subspaces, such explanations are usually a by-product of the outlier detection process. In addition, the subspaces identified in above methods are usually associated with all potential outliers, and the explanations offered by those subspaces may not be suitable for the query object that people are interested in.

Recent year, researchers began to focus on the explanations of the interested object, and existing methods can be divided into two categories, score-and-search based methods and feature selection based methods. To best of our knowledge, score-and-search based method was first introduced in [14]. They proposed an Euclidean distance base scoring function to measure the outlying degree of the query object in each candidate subspace, to identify the subspace in which the query object is most deviated from others. Later on, [4] replaced this Euclidean distance by a kernel density based scoring function, and [7, 12] proposed a data deviation based kernel density scoring function. Along another line of research, feature selection based methods offer the explanation for object interested by users. The terms of *outlier explanation* and *outlying aspects mining*, and most work in this category were originated from the series of work such as [8]. In this category, this object explanation problem is usually transformed into a feature selection based classification problem, in which the query object is defined as the positive class while the rest are considered as negative. In [10], the whole dataset is regarded as the negative class, and the positive class is over-sampled from the data.

However, all of the works mentioned above are focused on mining the outlying aspects of a single object, instead of getting insight into the outlying aspects of a group of objects, even though in many scenarios, it is essential to identify those aspects for a group.

3 Problem Definition

In this section, we illustrate the problem of *Group Outlying Aspects Mining*, and then formalize it with definitions which are helpful in guiding our subsequent algorithm development.

3.1 Problem Formalization

Suppose 3 basketball teams in a university: red team, blue team and green team. The players in those teams can be described by three features: **scoring**, **defence** and **organization**. If we focus on the red team, the *Group Outlying Aspects Mining* task could be the identification of a set of features in which the red team is most different from others. In other words, the potential questions in this context may be “Did the red team score better than the other teams?”, or “Did the red team score and defend better than the other teams?” or “Whether the red team is defensive or whether its organization is worse than the two teams?” et al.

Based the example above, we can formalize the *Group Outlying Aspects Mining* problem as follows:

Definition 1 (Group Outlying Aspects Mining). Let $G = \{G_q, G_2, G_3, \dots, G_n\}$ denote a set of groups, in which G_q is the query group, and $\{G_2, G_3, \dots, G_n\}$ are groups for comparison. Each object in the group has d features $F = \{f_1, f_2, \dots, f_d\}$. The group outlying aspects mining is to identify the non-empty subspace $s \subseteq F$ in which the query group G_q 's outlying degree is larger than a user specified threshold α .

3.2 Term Definitions

Intuitively, the query group could be different from the others directly on some raw features, or be different when considering several raw features together. Accordingly, these are corresponding to different subspaces, and we distinguish them by the following definitions:

Definition 2 (Trivial Outlying Features). Trivial outlying features are the raw features on which the query group is different from others, and they are defined as follows: let $\rho(\cdot)$ be a scoring function that measures the outlying degree of the query group G_q in a candidate subspace s , then the trivial outlying features are those features (one-dimension subspace) in which the query group G_q 's outlying degree $\rho(\cdot)$ is larger than the user specified threshold α .

Definition 3 (l-D Nontrivial Outlying Subspace). l-D ($l > 1$) nontrivial outlying subspace refers to the subspace consisting of l nontrivial outlying features but on which the query group G_q 's outlying degree $\rho(\cdot)$ is larger than the user specified threshold α .

Table 1. Notation table

Notation	Meaning
$\{G_q\}$	The query group
$\{G_2, \dots, G_n\}$	The n groups for comparison
$\{f_1, f_2, \dots, f_d\}$	The group and its members feature set
$\{G_1 = \{o_1, o_2, \dots, o_m\}\}$	The group G_1 with m members
$f_{i_{\setminus}} = \{f_{i_1}, f_{i_2}, \dots, f_{i_j}\}$	The value set of i -feature
H_i	The histogram representation of feature f_i
$ s $	The dimension of the possible subspaces
α	Threshold that used to determine the outlying aspects
d	The number of the features
$\rho(\cdot)$	The scoring function

Based on above definitions, the group outlying aspects mining task can be decomposed into two components: the identification of *Trivial outlying features* and the identification of highly scored *l -D Nontrivial outlying subspace*. It should be noted that, trivial outlying features should be examined and then filtered out before identifying nontrivial outlying subspaces, because coupling the trivial outlying features with other features will complicate the detecting of non-outlying features [12]. If we keep the trivial outlying features in the *group outlying aspects mining* process, the subspaces with highly outlying degree will likely be various combinations of trivial outlying features and other features, and this will result in missing some more interesting nontrivial subspaces. Notations used in this work are summarized in Table 1.

4 GOAM Algorithm

In this section, we present the GOAM algorithm in three major steps: (1) Group Feature Extraction, (2) Outlying Degree Scoring, and (3) Outlying Aspects Identification. The overall architecture of this algorithm is shown in Fig. 1.

4.1 Group Feature Extraction

Different with traditional *outlying aspects mining*, the target of *Group outlying aspect mining* is the group rather than a particular individual. Hence, we need to extract the features based on individual objects of the group. We use the histogram representation to achieve the purpose.

Suppose $G_i = \{o_1, o_2, \dots, o_m\}$ denotes a group of m objects, each of which is represented by d features $\{f_1, f_2, \dots, f_d\}$, and the values of feature f_i are $f_{i_{\setminus}} = \{f_{i_1}, f_{i_2}, \dots, f_{i_j}\}$, where f_{i_j} denotes the j -th value of feature f_i . It should be noted that, this representation requires each feature to be discrete, so continuous features need to be discretized. For each feature f_x of the group G_i , we can count

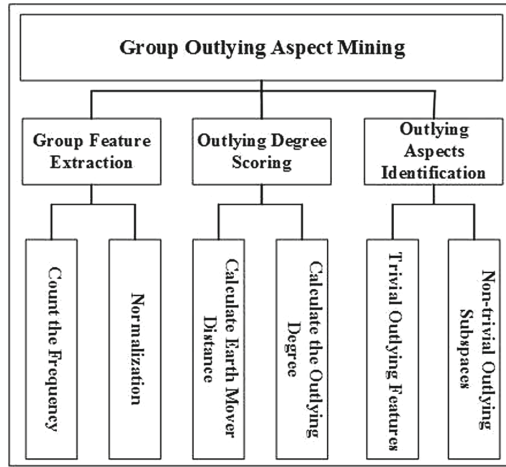


Fig. 1. Group outlying aspects mining framework

the frequencies of its values $f_{x_{\{j\}}} = \{f_{x_1}, f_{x_2}, \dots, f_{x_j}\}$ in the group, and normalize them as $H_x: \{[p_{x_1}, \dots, p_{x_j}]\}$, where p_{x_j} is the normalised frequency of f_{x_j} for the x -th feature f_x in the group.

Example 1. Suppose each member of the query group G_q can be represented by three features: f_1, f_2 and f_3 , and the values of objects in the group on those three features are: $\{x_1, x_2, x_3, x_4, x_5, x_2, x_3, x_4, x_1, x_2\}$, $\{y_2, y_2, y_1, y_2, y_3, y_3, y_5, y_4, y_4, y_2\}$ and $\{z_1, z_4, z_2, z_4, z_5, z_3, z_1, z_2, z_4, z_2\}$ respectively. The histogram representations for f_1, f_2 and f_3 of G_q are shown in Fig. 2.

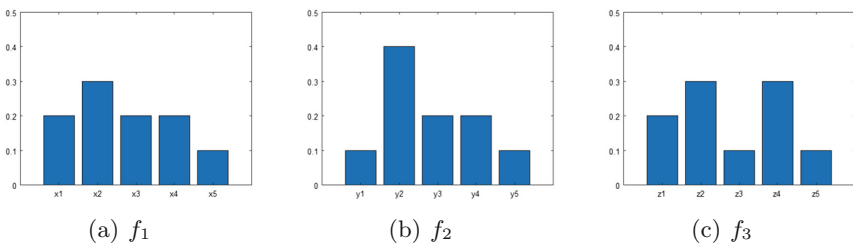


Fig. 2. Histogram representation of a group on three single features

4.2 Outlying Degree Scoring

After feature extraction, the group is characterized by a set of histograms as shown in Fig. 2. In order to evaluate the outlying degree between the query group and other groups on a candidate subspace. An appropriate scoring function $\rho(\cdot)$

that measures the difference of two groups on the candidate subspace is required. Considering that the group is represented essentially as a set of histograms, the *Earth Mover’s Distance* is adopted as the scoring function to evaluate the difference between groups.

Intuitively, the *Earth Mover’s Distance* of two histograms $h_1 : \{[p_{x_1}, \dots, p_{x_j}]\}$, $h_2 : \{[p_{x_1} \dots, p_{x_j}]\}$ is the minimum mean distance of moving the histogram h_1 into the histogram h_2 . It can be calculated as the following linear programming problem:

$$\min \sum_{i=1}^m \sum_{j=1}^n c_{i,j} d_{i,j} \quad \text{s.t.} \begin{cases} c_{i,j} \geq 0, & 1 \leq i \leq m, 1 \leq j \leq n, \\ \sum_{j=1}^n c_{i,j} \leq p_i, & 1 \leq i \leq m, \\ \sum_{i=1}^m c_{i,j} \leq q_j, & 1 \leq j \leq n, \\ \sum_{i=1}^m \sum_{j=1}^n c_{i,j} = \min\{\sum_{i=1}^m p_i, \sum_{j=1}^n q_j\}. \end{cases} \quad (1)$$

where $d_{i,j}$ represents the ground distance between h_1 and h_2 , $c_{i,j}$ denotes the move flow between h_1 and h_2 that minimizes the overall cost. The first constraint implies that the moving process is only one direction, and the second and third constraints ensure that the move “supplies” of h_1 and h_2 does not exceed the maximum amount, the last constrain ensures that the move of all available supplies is permitted. Once above linear programming problem is solved, the earth move distance between two histograms can be calculated according to Eq. (2).

$$EMD(h_1, h_2) = \frac{\sum_{i=1}^m \sum_{j=1}^n c_{i,j} d_{i,j}}{\sum_{i=1}^m \sum_{j=1}^n c_{i,j}} \quad (2)$$

Based on the EDM, the outlying degree between the query group G_q and other groups G_2, G_3, \dots, G_n in a subspace s can be calculated as:

$$OD(G_q) = \sum_1^n EDM(h_{q_s}, h_{k_s}) \quad (3)$$

where h_{k_s} is the histogram representation of G_k in the subspace s .

4.3 Outlying Aspects Identification

Once the scoring function is determined, in theory the identification process could be as trivial as using the scoring function to quantify the outlying degree of all candidate subspaces, followed by reporting the trivial outlying features and non-trivial outlying subspaces for user inspection. However, this naive strategy is with exponential complexity, because of $2^d - 1$ candidate subspaces. When the dimensionality d is high, the curse of dimensionality will be inevitable.

Here we adopt a stage-wise candidate subspace construction strategy to alleviate the exponential explosion by pre-filtering highly scored subspaces in each stage of candidate subspace construction. The framework of the GOAM is presented in Algorithm 1, where d represents the number of features in the original data space. The detailed Algorithm 1 can be divided into three parts, the first part is step 1 which identifies the trivial outlying features, the second part is the loop between step 2 and step 9 which generates and identifies the nontrivial outlying subspace for each dimensionality, the last part is step 10 which identifies the final nontrivial outlying subspace.

Algorithm 1. The Framework of GOAM

Require: group data $G_1, G_2, G_3, \dots, G_n$.

Ensure: the outlying aspects of G_1

$T_1 = \{f_i | \rho(f_i) \geq \alpha\}$. {Find trivial outlying features}

for $|l| = 1, |l| \leq d, |l|++$ **do**

$C_{l+1} = \text{Generate Candidate Subspaces}(N_l)$. {Generate $l+1$ -D candidate subspace}

for each subspaces in the candidate list C_{l+1} **do**

calculate the outlying degree.

end for

$N_{l+1} =$ subspaces in C_{s+1} with outlying degree bigger than α . {Find $l+1$ -D nontrivial outlying subspaces}

Add N_{l+1} to C_r , and pruning N_{l+1} leaving the rest of subspace in C_{s+1} participate following construction stage.

end for

Sort C_r in descending order, add subspace with top- K outlying degree in C_r to N_r .

return N_r, T_1

Similar to the Apriori algorithm, the GOAM algorithm achieves the efficiency through a complete bottom-up search strategy which reduces the size of candidate subspaces. The naive method is with the time complexity of $O(2^d)$, while in GOAM, the time complexity is only $O(d \times n^2)$.

5 Experiment and Analysis

This section presents a series of experiments to demonstrate the effectiveness of the proposed method in identifying the outlying aspects of groups. All experiments were performed on an i7 – 6700 quad-core desktop PC with 8 GB RAM. The parameter setting in our experiments are $\alpha = 4.0$, and $K = 2$.

5.1 Mining Group Outlying Aspects on Synthetic Dataset

First we conduct a series of experiments on a synthetic dataset with the ground truth of outlying aspects to illustrate the effective of the proposed method.

The dataset we used in our experiment contains 10 groups, each of which consists of 10 members, and each member is with 8 features $\{F_1, F_2, F_3, \dots, F_8\}$.

Table 2. The query group

Query group	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8
i_1	10	8	9	7	7	6	6	8
i_2	9	9	7	8	9	9	8	9
i_3	8	10	8	9	6	8	7	8
i_4	8	8	6	7	8	8	6	7
i_5	9	9	9	7	7	7	8	8
i_6	8	10	8	8	6	6	8	7
i_7	9	9	7	9	8	8	8	7
i_8	10	9	10	7	7	7	7	7
i_9	9	10	8	8	7	6	7	7
i_{10}	9	9	7	7	7	8	8	8

Table 3. The experiment result on synthetic dataset

Method	Truth outlying aspects	Identified aspects	Accuracy
GOAM	$\{F_1\}, \{F_2F_4\}$	$\{F_1\}, \{F_2F_4\}$	100%
Arithmetic mean based OAM	$\{F_1\}, \{F_2F_4\}$	$\{F_4\}, \{F_2\}$	0%
Median based OAM	$\{F_1\}, \{F_2F_4\}$	$\{F_2\}, \{F_4\}$	0%

This part makes use of the thought of [6] and use the algorithm to synthesize the data sets for the group outlying aspects mining. Table 2 shows the original data of the query group in our dataset, and the ground truth outlying aspects in our dataset includes the *trivial outlying feature* $\{F_1\}$ and the *non-trivial outlying subspace* $\{F_2, F_4\}$. In $\{F_1\}$, we can see that, the histogram of the query group is concentrated on 8 to 10, while other groups are 6 to 8. The histogram of the query group in $\{F_2, F_4\}$ is high around $\{9, 9\}$, while other groups are high on $\{5, 6\}$ and $\{5, 7\}$.

Let those ground truth outlying aspects be T and the identified outlying aspects be P . The accuracy $Acc = \frac{|P|}{|T|}$ was used to evaluate the effectiveness of the algorithms. We compared our algorithm with the state of art *outlying aspect mining* method [12]. The experiment results are shown in Table 3.

From the Table 3, we can see that GOAM can identify both of *trivial outlying features* and *nontrivial outlying features* correctively, its algorithm accuracy is 100%, while the state of art *outlying aspects mining* based method fail to identify them for groups effectively, all the corresponding algorithm accuracy are 0%. This is due to the fact that *outlying aspects mining* based method does not capture the group features, and the scoring function is based on a point-point metric, which is not suitable for the group level.

5.2 Mining Group Outlying Aspects on Real Dataset

In this subsection, we apply the proposed GOAM algorithm on a real world dataset to demonstrate its capability in identify the group outlying aspects.

The data was collected from *Yahoo Sports* (<http://sports.yahoo.com.cn/nba>), one online platform that provides professional sport news and events coverage. *Yahoo Sports* has been commonly used as the data source in outlying aspect mining, such as [4, 11], and it provides detailed data of all NBA teams and players from 2002/2003 to the present. In our experiment, a web crawler was deployed to extract data for all NBA teams until March 30, 2018. Players who have not played in that season because of injury or tactics reasons are excluded in our data collection process, and features with little use for the analysis of team characteristics are excluded.

Table 4 shows part of the data collected by *Brooklyn Nets* team, where *Pts* denotes the point scored, *FGA* denotes the field goal attempts, *FG%* denotes the field goal percentage, *3FA* denotes the 3-point shots attempted, *3PT%* denotes the 3-point percentage, *FTA* denotes free throws attempted, *FT%* denotes the free throw percentage, *Reb* denotes the total rebounds, *Ass* denotes the assists, *To* denotes the turnovers, *Stl* denotes the steals, *Blk* denotes the blocked shots.

For those features with continuous value, we use the binning method to discretize them, as shown in Table 5. Once the data were prepared, we use three teams in the eastern (*Cleveland Cavaliers*, *Orlando Magic*, *Milwaukee Bucks*) and the western league (*Golden State Warriors*, *Utah Jazz*, *New Orleans Pelicans*) respectively as the query group, the other teams in the corresponding

Table 4. Collected data of brooklyn nets team

Pts	FGA	FG%	3FA	3PT%	FTA	FT%	Reb	Ass	To	Stl	Blk
18	12	0.42	2.00	0.50	7.00	1.00	0	4	3	0	0
15.7	14.07	0.41	5.45	0.32	3.05	0.75	3.98	5.1	2.98	0.69	0.36
14.5	11.1	0.47	0.82	0.26	4.87	0.78	6.82	2.4	1.74	0.92	0.66
13.5	10.8	0.42	5.37	0.37	3.38	0.77	6.66	2	1.38	0.83	0.42
12.7	10.59	0.39	5.36	0.33	3.37	0.82	3.24	6.6	1.56	0.89	0.31
12.6	10.93	0.40	6.94	0.37	1.70	0.84	4.27	1.5	1.06	0.61	0.44
12.2	10.39	0.44	3.42	0.35	2.70	0.72	3.79	4.1	2.15	1.12	0.32
10.6	7.85	0.49	4.51	0.41	1.35	0.83	3.34	1.6	1.15	0.45	0.24

Table 5. The bins that used to discrete data of each feature

Labels	Pts	FGA	FG%	3FA	3PT%	FTA	FT%	Reb	Ass	To	Stl	Blk
low	[0,5]	[0,4]	[0,0.35]	[0,1.0]	[0,0.2]	[0,1.0]	[0,0.6]	[0,2.0]	[0,1.0]	[0,0.6]	[0,0.2]	[0,0.25]
medium	(5,10]	(4,7]	(0.35,0.45]	(1.0,2.5]	(0.2,0.3]	(1.0,1.5]	(0.6,0.65]	(2,5]	(1,2]	(0.6,0.9]	(0.2,0.5]	(0.25,0.5]
high	(10,15]	(7,10]	(0.45,0.5]	(2.5,3.5]	(0.3,0.35]	(1.5,2.5]	(0.65,0.75]	(5,6]	(2,4]	(0.9,1.7]	(0.6,0.75]	(0.5,0.7]
very high	(15,+∞)	(10,+∞)	(0.5,1]	(3.5,+∞)	(0.35,1]	(2.5,+∞)	(0.75,1]	(6,+∞)	(4,+∞)	(1.7,+∞)	(0.75,+∞)	(0.7,+∞)

Table 6. The identified outlying aspects of groups

Teams	Trivial outlying aspects	Nontrivial outlying aspects
Cleveland cavaliers	{3FA}	{FGA, FT%}, {FGA, FG%}
Orlando magic	{Stl}	None
Milwaukee bucks	{To}, {FTA}	{FGA, FTA}, {3FA, FTA}
Golden state warriors	{FG%}	{FT%, Blk}, {FGA, 3PT%, FTA}
Utah jazz	{Blk}	{3FA, 3PT%}
New orleans pelicans	{FT%}, {FTA}	{FTA, Stl}, {FTA, To}

partition as the contrast groups to demonstrate the capability of mining the outlying aspects of group by the proposed method. The experiment results are shown in Table 6.

From Table 6, we can find that the most different features/subspace of **Cleveland Cavaliers** are from its {3-point attempt}, {free throw percentage, field goal attempts} and {field goal attempts, field goal percentage}. When only focusing on {3-point attempt}, {free throw percentage, field goal attempts} and {field goal attempts, field goal percentage}, we can see that comparing with other teams in eastern league, **Cleveland Cavaliers** has more three-point pitchers, more of its player with higher ratings of *free throw percentage* and *field goal attempts* at the same time, or with higher ratings of *field goal attempts, field goal percentage* at the same time. Similarly, we can see that **Orlando Magic** has more players who are good at {steals} than their counterparts in the east league. **Milwaukee Bucks** has more players have fewer {turnovers} and more {free throws attempted} than other teams in the east league. **Milwaukee Bucks** has more players with lower scores on {free throws attempted, field goal attempts} and {3-point shots attempted, free throws attempted} than other teams in the east league. **Golden State Warriors** has more players who are good at {field goal percentage}, {free throw percentage, blocked shots} and {field goal attempts, 3-point percentage, free throws attempted} than other teams in west league. **Utah Jazz** has more players are good at {blocked shots} than other teams in west league. At the same time, **Utah Jazz** has more players who are not good at {3-point percentage, 3-point shots attempted} than other teams in west league. **New Orleans Pelicans** has more players with lower {free throws attempted}, {free throw percentage}, {turnovers, free throws attempted} and {free throws attempted, steals} than other teams in west league.

6 Discussion and Conclusions

The knowledge of outlying aspects of a particular group is valuable in many decision making practices. If the practitioners can know these patterns, suitable marketing or improvement strategy will be made based on evidences. However,

most of existing works on outlying aspects mining pay attention to identifying the outlying aspects of individual object rather than a group. It lacks of methods that can effectively mine the outlying aspects of a particular group. To fill this research gap, this paper extends the traditional problem to the group level, and formalizes the concept of *group outlying aspect mining*. The Earth Move Distance based algorithm GOAM is proposed in Sect. 4 to automatically identify the outlying aspects of the query group. The experiments on Sect. 5 demonstrated the effectiveness of the proposed methods in mining these patterns.

Acknowledgement. This work was supported by the International Cooperation Project of Institute of Information Engineering, Chinese Academy of Sciences under Grant No. Y7Z0511101, and also supported by the practical training project of high-level talents cross training of Beijing colleges and universities (BUCEA-2016-28).

References

1. Aggarwal, C.C., Yu, P.S.: Outlier detection for high dimensional data. ACM SIGMOD Record, vol. 30, no. 2, pp. 37–46 (2001)
2. Dang, X.H., Assent, I., Ng, R.T., Zimek, A., Schubert, E.: Discriminative features for identifying and interpreting outliers. In: IEEE International Conference on Data Engineering, pp. 88–99 (2014)
3. Dang, X.H., Micenková, B., Assent, I., Ng, R.T.: Local Outlier Detection with Interpretation. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-40994-3_20
4. Duan, L., Tang, G., Pei, J., Campbell, A., Campbell, A., Tang, C.: Mining outlying aspects on numeric data. Data Min. Knowl. Discov. **29**(5), 1116–1151 (2015)
5. Kriegel, H.P., Hubert, M.S., Zimek, A.: Angle-based outlier detection in high-dimensional data. In: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 444–452 (2008)
6. Li, Q., Li, G., Niu, W., Cao, Y., Chang, L., Tan, J., Guo, L.: Boosting imbalanced data learning with wiener process oversampling. Front. Comput. Sci. **11**(5), 836–851 (2017)
7. Li, Q., Niu, W., Li, G., Cao, Y., Tan, J., Guo, L.: Lingo: linearized grassmannian optimization for nuclear norm minimization. In: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, pp. 801–809. ACM (2015)
8. Micenkova, B., Dang, X.H., Assent, I., Ng, R.T.: Explaining outliers by subspace separability. In: IEEE International Conference on Data Mining, pp. 518–527 (2013)
9. Nguyen, H.V., Muller, E., Vreeken, J., Keller, F., Bohm, K.: CMI: an information-theoretic contrast measure for enhancing subspace cluster and outlier detection, pp. 198–206 (2013)
10. Vinh, N.X., Chan, J., Bailey, J.: Reconsidering mutual information based feature selection: a statistical significance view. In: IEEE International Conference on Embedded and Real-Time Computing Systems and Applications, pp. 1–10 (2014)
11. Vinh, N.X., Chan, J., Bailey, J., Leckie, C., Ramamohanarao, K., Pei, J.: Scalable outlying-inlying aspects discovery via feature ranking. In: IEEE International Symposium on Biomedical Imaging, pp. 182–185 (2015)

12. Vinh, N.X., Chan, J., Romano, S., Bailey, J., Leckie, C., Ramamohanarao, K., Pei, J.: Discovering outlying aspects in large datasets. *Data Min. Knowl. Discov.* **30**(6), 1520–1555 (2016)
13. Xiong, P., Wang, X., Niu, W., Zhu, T., Li, G.: Android malware detection with contrasting permission patterns. *China Commun.* **11**(8), 1–14 (2014)
14. Zhang, J., Lou, M., Ling, T.W., Wang, H.: HOS-miner : a system for detecting outlying subspaces of high-dimensional data. In: Thirtieth International Conference on Very Large Data Bases, pp. 1265–1268 (2004)



Fine-Grained Correlation Learning with Stacked Co-attention Networks for Cross-Modal Information Retrieval

Yuhang Lu^{1,2}, Jing Yu^{1(✉)}, Yanbing Liu¹, Jianlong Tan¹, Li Guo¹,
and Weifeng Zhang^{3,4}

¹ Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
{luyuhang,yujing02,liuyanbing,tanjianlong,guoli}@iie.ac.cn

² School of Cyber Security, University of Chinese Academy of Sciences,
Beijing, China

³ School of Computer Science and Technology, Hangzhou Dianzi University,
Hangzhou, China

zwf.zhang@gmail.com

⁴ Zhejiang Future Technology Institute, Jiaxing, China

Abstract. Cross-modal retrieval provides a flexible way to find semantically relevant information across different modalities given a query of one modality. The main challenge is to measure the similarity between different modalities of data. Generally, different modalities contain unequal amount of information when describing the same semantics. For example, textual descriptions often contain more background information that cannot be conveyed by images and vice versa. Existing works mostly map the global data features from different modalities to a common semantic space to measure their similarity, which ignore their imbalanced and complementary relationships. In this paper, we propose stacked co-attention networks (SCANet) to progressively learn the mutually attended features of different modalities and leverage these fine-grained correlations to enhance cross-modal retrieval performance. SCANet adopts a dual-path end-to-end framework to jointly learn the multimodal representations, stacked co-attention, and similarity metric. Experiment results on three widely-used benchmark datasets verify that SCANet outperforms state-of-the-art methods, with 19% improvements on MAP in average for the best case.

Keywords: Stacked co-attention network · Graph convolution
Fine-grained cross-modal correlation

1 Introduction

With the fast development of Internet and mobile network, multimodal data including image, text, video and audio, has been emerging and accumulated rapidly. Multimedia retrieval becomes a fundamental technique for intelligent

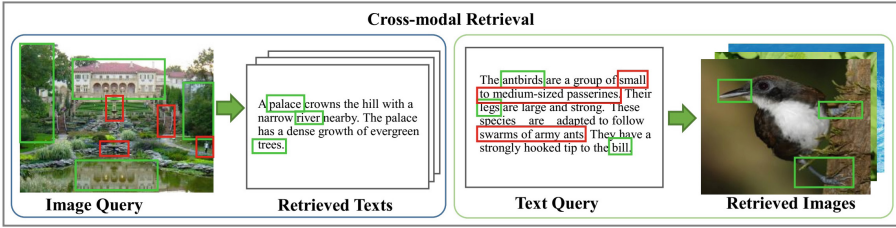


Fig. 1. Examples of image-text retrieval. The samples of different modalities have imbalanced and unequal information. The green boxes indicate related content appeared in both modalities while the red boxes mean extra content existed in only one modality.

search and big data analysis. Traditional information retrieval approaches, such as key word retrieval and content-based image retrieval, belong to single-modal models only obtaining results of the same modality with the query. However, such models limit the flexibility for accessing various modalities of data. Cross-modal retrieval provides semantic relevant information from multimodal data when given a query of one modality. This is much closer to what humans do than matching text or visual features independently.

The main challenge for cross-modal retrieval is to bridge the “heterogeneous gap” between different modalities and learn an appropriate similarity metric between them. A common solution of existing work is to map the features of different modalities into one common latent space and compare their similarity by the learnt common representations. Conventional methods are based on Canonical Correlation Analysis (CCA) [15], probabilistic models [1], metric learning [24], and joint modeling [18]. Recently, Deep Neural Network (DNN) has promoted the advances in cross-modal retrieval, such as deep feature representation with CNN [6] and LSTM, common space learning with multilayer network, and similarity measure with DNN-based metric learning.

The aforementioned approaches map the complete data of different modalities from their feature space to the common semantic space equally to find their feature correlations, which is based on the assumption that semantically relevant data of different modalities has equal amount of information. However, this assumption is not always true in practice. In fact, data conveying the same semantics but from different modalities may express imbalanced information and have complementary relationships. For example, an image is usually accompanied by a text description and vice versa to express the same semantics, but the amount of information between the image and the text is unequal. In Fig. 1, the left image query contains a complex scenery which cannot be completely described by a few objects in the retrieved texts. On the other scenarios, the right query text has more background information beyond the content of only an antbird in the retrieved images. It’s not all the fine-grained information between text and image has exact correlations. Therefore, regarding all different modalities equally will weaken some important aligned information while introducing unaligned noise.

Resently, Peng *et al.* [12] has demonstrated the advantages of fine-grained information in modality-specific space for cross-modal correlation modeling. Different from their work, we focus on preserving the mutual fine-grained parts of different modalities in the common semantic space to learn the cross-modal correlations. Inspired by the progress of attention mechanism in image caption and visual question answering, we propose **Stacked Co-Attention Networks** (SCANet) to explore the mutually attended characteristics of different modalities for strengthening the semantic correlations.

The rest of this paper is organized as follows. We briefly review the related works in Sect. 2. Section 3 introduces our proposed SCANet approach. We report the experimental results in Sect. 4 and conclude our work in Sect. 5.

2 Related Work

DNN-Based Feature Extraction. Feature representation is the footstone for cross-modal retrieval. In the text-image retrieval field, off-the-shelf features learnt by deep neural networks are widely utilized to represent images. Most existing works use Convolutional Neural Network (CNN) pre-trained on ImageNet to extract visual features for text-image semantic space mapping. However, such CNN model is pre-trained for object recognition which may ignore some detailed information for other tasks. Therefore, fine-tuning off-the-shelf CNN features for more discriminative embeddings is necessary for cross-modal-specific tasks.

For text representation, the popular vector-space models are usually used to convert a text to a high-level semantic vector based on the sequential word embeddings. Recurrent Neural Networks (RNN) is one of the popular choices in this kind of models. Nam *et al.* [10] applies directional LSTM for text representation and results in remarkable multimodal retrieval accuracy. Peng *et al.* [11] utilizes attention-based LSTM for modality-specific feature learning to refine the cross-modal correlations. Meanwhile, CNN-based text modeling also yields competitive results in image-sentence retrieval. These vector-space models treat the input words as “flat” features and ignore the global semantic structures inherent in the text. Recent research has found that the relations among words could provide rich semantics. Graph Convolutional Network (GCN) [7] is one popular graph-based neural network and has been used to model the semantic relations in a text as a featured graph. It has a great ability to learn local and stationary features and can effectively promote the text classification performance. In this paper, in stead of RNN which is commonly used in text-image retrieval, we explore the usage of GCN for text feature extraction.

Cross-Modal Learning. The mainstream solution for cross-modal retrieval is to project the features of different modalities into a common semantic space and measure their similarity directly. The traditional statistical correlation analysis methods, typically like Canonical Correlation Analysis (CCA) [15], aim to maximize the pairwise correlations between two sets the data of different modalities. In order to leverage the semantic information, graph-based semi-supervised methods [18] and supervised methods are proposed to explore the label

information and achieve great progress. With the advances of deep learning in multimedia applications, DNN-based cross-modal methods are in the ascendant. This kind of methods generally construct two subnetworks for modeling data of different modalities and learn their correlations by a joint layer. Zheng *et al.* [25] uses two convolution networks for learning textual-visual embeddings and realize effective end-to-end fine-tuning. In this work, we also follow the DNN-based routine to model the matched and mismatched text-image pairs.

Attention Mechanism. Recently, attention mechanism has promoted remarkable advances in many multimodal tasks, such as image caption, image question answering, cross-modal retrieval and etc. It allows deep models to focus on the task-driven necessary parts of the features. Yang *et al.* [20] proposes Stacked Attention Networks (SANs), which takes multiple attention steps to progressively focus on the informative parts for image question answering. Attention-based cross-modal retrieval models aim to simultaneously locate the necessary components in both textual and image features to learn more accurate semantic correlations. For example, Zhang *et al.* [23] generates adaptive attention masks and divides features into attended and unattended parts to enhance the robustness of learnt representations. Peng *et al.* [12] designs a recurrent attention network to capture the modality-specific characteristics in textual and image space independently. Different from their work, we use attention mechanism to fully explore the co-attended parts inherent in both of the two modalities for learning better cross-modal correlations.

3 Methodology

In this section, we present **Stacked Co-attention Netowrks** (SCANet) using dual-path neural networks. The overall architecture of SCANet is shown in Fig. 2. We describe the major four components of SCANet: the text model, the image model, the stacked co-attention network, and the objective function.

3.1 Text Model

As illustrated in [7], Graph Convolutional Network (GCN) shows strong ability in modeling the semantics of texts and has good performance in text classification [2]. We explore GCN to learn the text features to leverage the prior information of semantic similarities inherent in the text corpus. We first represent each text by a featured graph as the input of GCN. The graph structure is identical for all the texts while the graph features are unique for each text. We extract the most common words from all the unique words in the text corpus and represent each word by a pre-trained *word2vec* [9] embedding. Then each vertex in the graph structure is corresponding to a common word. For each vertex, we compute its k -nearest neighbors of vertices based on the cosine similarity between word *word2vec* embeddings to form the edge set. For the graph features, each text is represented by a *bag-of-words* vector and the word frequency serves as the 1-dimensional feature on the corresponding word vertex.

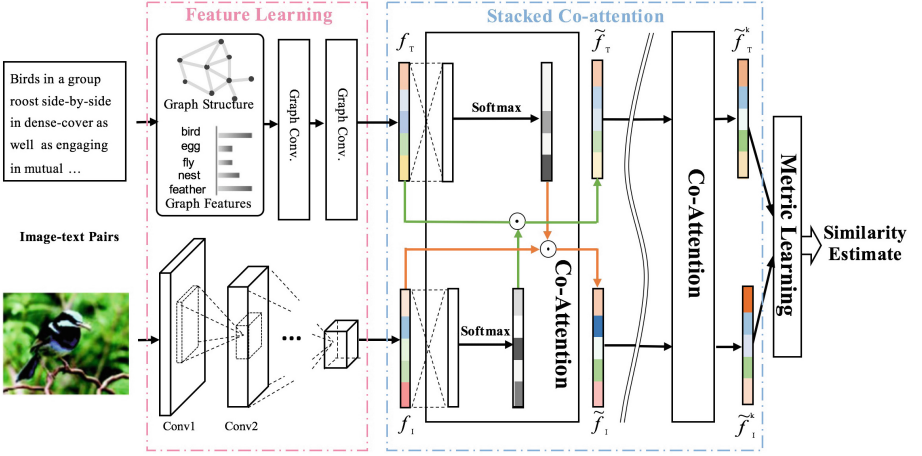


Fig. 2. Overview of SCANet. Above is the text modality path and below is the image modality path. Each path is divided into two parts: feature learning (f_T and f_I) and stacked co-attention (f_T^k and f_I^k). The feature learning maps each modality data into high-level feature representation with the same dimension. The stacked co-attention conducts co-attention process multiple times. For each co-attention process, the attention distribution learnt from image features are used to update the text features and vice versa. Finally, the two paths are joint by metric learning for the similarity measure.

Given the featured graph of each text, we learn the text features using GCN [7]. The input and output features are defined by F_{in} and F_{out} . The i th output feature $f_{out,i} \in F_{out}$ corresponding to the i th input feature $f_{in,i}$ is given by:

$$f_{out,i} = \sum_{k=0}^{K-1} \theta_k T_k(\tilde{L}) f_{in,i} \quad (1)$$

where θ_k is the parameter to learn and $T_k(\tilde{L}) = 2\tilde{L}T_{k-1}(\tilde{L}) - T_{k-2}(\tilde{L})$ with $T_0(\tilde{L}) = 1$ and $T_1(\tilde{L}) = \tilde{L}$. $\tilde{L} = \frac{2}{\lambda_{max}}L - I_N$ and λ_{max} denotes the largest eigenvalue of the normalized graph Laplacian L of the input graph structure. K is set to 3. In our model, GCN contains two layers of graph convolutions, each followed by Rectified Linear Unit (ReLU) activation to increase non-linearity. Then a fully connected layer is used to reduce the feature dimension and map the text to the common latent space with the image. Given a text T , the text representation f_T learnt by the GCN model $H_T(\cdot)$ is denoted by:

$$f_T = H_T(T) \quad (2)$$

3.2 Image Model

For modeling images, we use CNN to get the embeddings of images. Specially, we use the pre-trained VGGNet [17] and extract the features from last fully

connected layers as the image features. After getting the fixed VGG-19 image features, we use a set of fully connected (FC) layers for fine-tuning the feature representations. Similar to the text model, the last FC layer is used to reduce the feature dimension and map the image to the common latent space with the text. In the experimental study, we tune the number of FC layers and find that only keeping the last common space mapping layer without extra feature fine-tuning can obtain the best retrieval performance. Given an image I , the image representation f_I learnt by the image model $H_I(\cdot)$ is demoted by:

$$f_I = H_I(I) \quad (3)$$

3.3 Stacked Co-attention Networks

Given the text features f_T and the image features f_I , the stacked co-attention networks learn the fine-grained semantic correlations between these two modalities via multiple attention layers. In many cases, the amount of information in different modalities is unequal even if they convey the same semantics. Using the global image features and text features for feature alignment could introduce noise from their irrelevant parts. Therefore, the stacked co-attention networks are proposed to progressively attend to the parts that are highly correlated between image and text pairs and gradually filter out the unaligned noise.

As shown in Fig. 2, the inputs of stacked co-attention networks the are features f_T and f_I of two modalities. In the first co-attention layer, the text features f_T is updated by the attention distribution learnt from the image features and we denote the output text features as \tilde{f}_T . While the image features f_I is updated by the attention distribution learnt from the text features and we denote the output image features as \tilde{f}_I . Specifically, given f_T and f_I , we put them into a fully connected layer and then use a *softmax* function to generate the attention distribution. Here is the formula for computing the attention distribution:

$$h_T = \text{relu}(W_T f_T + b_T) \quad p_T = \text{softmax}(h_T) \quad (4)$$

$$h_I = \text{relu}(W_I f_I + b_I) \quad p_I = \text{softmax}(h_I) \quad (5)$$

where $f_T \in \mathbb{R}^N$, $f_I \in \mathbb{R}^N$, $W_T \in \mathbb{R}^{N \times N}$, $W_I \in \mathbb{R}^{N \times N}$, $b_T \in \mathbb{R}^N$ and $b_I \in \mathbb{R}^N$. p_I is the attention distribution learnt from the image feature while p_T is the attention distribution learnt from the text feature. Based on the attention distribution, we calculate the weighted sum of the text features and image features respectively and get the updated features denoted as \tilde{f}_T and \tilde{f}_I according to the following formula:

$$\tilde{f}_T = p_I \circ f_T \quad \tilde{f}_I = p_T \circ f_I \quad (6)$$

where “ \circ ” is Hadamard product. \tilde{f}_T and \tilde{f}_I are the inputs of the second co-attention layer. The co-attention layer will repeat multiple times to gradually obtain the fine-grained correlations. We denote the outputs of the k -th co-attention layer as \tilde{f}_T^k and \tilde{f}_I^k for the text and image respectively. Formally, the k -th co-attention layer is computed by the following formula:

$$h_T^k = \text{relu}(W_T^k f_T^{k-1} + b_T^k) \quad p_T^k = \text{softmax}(h_T^k) \quad (7)$$

$$h_I^k = \text{relu}(W_I^k f_I^{k-1} + b_I^k) \quad p_I^k = \text{softmax}(h_I^k) \quad (8)$$

Based on the attention distribution p_T^k and p_I^k , we can get the updated text features \tilde{f}_T^k and image features \tilde{f}_I^k by the following formula:

$$\tilde{f}_T^k = p_I^k \circ f_T^{k-1} \quad \tilde{f}_I^k = p_T^k \circ f_I^{k-1} \quad (9)$$

The number of co-attention layers depends on the characteristics of the multimodal data. If the semantics of the data are very complex, we need more co-attention layers to achieve better results. However, it should be noted that different pairs of multimodal data have different degree of information imbalance, and the number of co-attention layers needs to adapt to the specific data. Too many or too few layers may reduce the overall performance. In our experiments, using two co-attention layers is often the best choice.

3.4 Objective Function

In our model, we have finally get two modal features from the two path. Distance metric learning is applied to estimate the relevance of the two features. An inner product layer combines the two features and is followed by a single fully connected output layer with a sigmoid activation function and one output, that is the similarity we measure the two features. And the training objective is a pairwise similarity loss function proposed in [8]. The main idea is that we maximize the mean similarity score u^+ between text-image pairs of the same semantic concepts and minimize the mean similarity score u^- between pairs of different semantic concepts. Meanwhile, we also minimize the variance of matching pairs' similarity scores σ^{2+} and non-matching pairs' similarity scores σ^{2-} . We can get the *loss* by:

$$Loss = (\sigma^{2+} + \sigma^{2-}) + \lambda \max(0, m - (u^+ - u^-)) \quad (10)$$

where λ can adjust the proportion of mean, and m controls the upper limit between the mean of matching and non-matching similarity.

4 Experiments

To exhaustively evaluate our model, we conduct experiments on five benchmark datasets to compare our model with 11 state-of-the-art methods. Besides, we implement several baseline models to explore how variation in text features, image features, and the number of co-attention layers affects the performance of the proposed co-attention model. What's more, we also report the sensitive analysis of hyper-parameters including m and λ in the loss function. All the experiments are implemented by Tensorflow-1.6.0 using Python-3.5.0.

Table 1. Descriptions of the three benchmark datasets.

Dataset	#Training	#Testing	#Category	Image features	Text features
Eng-Wiki	2173 (original)	693	10	VGG-4,096	GCN-10,055
NUS-WIDE	40,000 (original)	1,000	10	BOF-500	GCN-5,018
Pascal VOC	2,808 (original)	2,841	20	Gist-512	GCN-598

4.1 Datasets

We use four popular datasets for our experiments, including English Wikipedia (Eng-Wiki for short), NUS-WIDE, Pascal VOC, and TVGraz. The detailed information is listed in Table 1. Based on the original training image-text pairs, we randomly select matched and non-matched pairs and form 40,000 positive samples and 40,000 negative samples for model training. The preprocessing methods for the datasets are the same as [21]. We preprocess the texts of each dataset to construct a featured graph with different vertex number as the input of GCN. We set $k = 8$ in k -nearest neighbors for text graph construction. The learnt text representation by GCN has the same dimension with the input feature. For image features, we adopt 4,096-dimensional VGG-19 [17] features for Eng-Wiki. Since NUS-WIDE and Pascal VOC datasets haven't provided the original images online, we use the off-the-shelf 500-dimensional bag-of-features and 512-dimensional Gist features, respectively.

4.2 Evaluation Measure and Experimental Settings

The mean average precision (MAP) [12] is used for experimental evaluation. Higher MAP indicates better retrieval performance. We train the model for 80 epochs with mini-batch size 256. We adopt the dropout ratio of 0.2 at the input of the last FC layer, learning rate 0.0001 with an Adam optimisation, and regularisation 0.005. m and λ in the loss function are set to 0.6 and 0.35, respectively. In the preceding layer of stacked co-attention, the text and image features are reduced to the same dimensions, which are set to 1,024, 500, 256 for En-Wiki, NUS-WIDE, Pascal, respectively.

4.3 Comparison with State-of-the-Art Methods

We first compare our model with several state-of-the-art methods, including CCA & SCM [15], TCM [13], GMLDA & GMMFA [16], LCFS [19], MvDA [4], LGCFL [5], ml-CCA [14], AUSL [22] and JFSSL [18]. All these methods are well cited work in this field. Since not all the papers have tested on the three datasets, for fair comparison, we compare our model to methods on their reported datasets with the same preprocessing conditions.

From Table 2 we can see that, for the text query task, SCANet outperforms all the other methods on all the datasets. Compared with the second best models, SCANet obtains remarkable improvements in MAP by 40%, 20% and 20% on

Table 2. Comparisons of MAP with state-of-the-art methods on three datasets.

Method	Eng-Wiki			NUS-WIDE			Pascal VOC		
	Text	Image	Average	Text	Image	Average	Text	Image	Average
SCM [15]	0.23	0.28	0.26	-	-	-	-	-	-
TCM [13]	0.29	0.23	0.26	-	-	-	-	-	-
CCA [15]	0.19	0.22	0.20	0.27	0.29	0.28	0.22	0.27	0.24
LCFS [19]	0.20	0.27	0.24	0.34	0.47	0.41	0.27	0.34	0.31
MvDA [4]	0.23	0.30	0.26	-	-	-	-	-	-
LGCFL [5]	0.32	0.38	0.35	0.39	0.50	0.44	-	-	-
ml-CCA [14]	0.29	0.35	0.32	0.39	0.47	0.43	-	-	-
GMLDA [16]	0.29	0.32	0.30	0.24	0.31	0.28	-	-	-
GMMFA [16]	0.30	0.32	0.31	0.23	0.31	0.27	-	-	-
AUSL [22]	0.33	0.40	0.37	0.41	0.57	0.49	-	-	-
JFSSL [18]	0.41	0.47	0.44	0.38	0.40	0.39	0.28	0.36	0.32
SCANet	0.81	0.46	0.63	0.58	0.52	0.55	0.48	0.34	0.41

Eng-Wiki, NUS-WIDE and Pascal VOC respectively. It proves that SCANet can represent the text well no matter for rich texts such as Eng-Wiki, or for sparse texts such as NUS-WIDE and Pascal VOC. It strongly proves the effectiveness of our proposed stacked co-attention model. The improvement also owes to the text GCN model, which leverages the semantic similarity structure inherent in the text corpus as a prior knowledge to progressively enhance the feature learning. The text embeddings learnt by GCN have good generalization ability.

For the image query task, the MAP of SCANet is superior to most of the compared methods. SCANet ranks second best on all the datasets, which is slightly inferior than the best performance. It is because for the image model of SCANet, we use pre-trained deep model VGGNet [17], which is trained for image classification task, to extract the image features. In future work, we may consider connecting the VGGNet with our model for end-to-end training. The main contribution of this paper doesn't focus on the image feature extraction. For the average performance, SCANet has 19%, 16%, and 9% improvements compared with the second best results on Eng-Wiki, NUS-WIDE and Pascal VOC respectively. The improvements are different on these datasets, which is also related to the image features. Eng-Wiki provides original images and we use VGG-19 to extract image features. While NUS-WIDE and Pascal VOC doesn't provide original images, we just use 500-dimensional bag-of-features and 512-dimensional Gist features respectively as they provided.

4.4 Baseline Comparisons

Besides our proposed model, we implement another four baseline models to evaluate the influence of the variation in text features and image features on the

Table 3. Comparisons of MAP with five baseline methods w.r.t different text features, image features, and co-attention layer numbers.

Text features	Image features	Text query				Image query				Average			
		#Attention layers				#Attention layers				#Attention layers			
		0	1	2	3	0	1	2	3	0	1	2	3
LSTM	Fixed VGG-19	0.62	0.64	0.63	0.60	0.42	0.44	0.43	0.42	0.52	0.54	0.53	0.51
CNN	Fixed VGG-19	0.36	0.37	0.41	0.35	0.30	0.31	0.33	0.29	0.33	0.34	0.37	0.32
GCN	Fixed VGG-19	0.75	0.76	0.81	0.72	0.43	0.45	0.46	0.46	0.59	0.61	0.63	0.59
GCN	Fixed ResNet-50	0.66	0.69	0.71	0.70	0.39	0.41	0.43	0.40	0.53	0.56	0.57	0.55
GCN	CNN-5	0.28	0.31	0.33	0.30	0.27	0.28	0.28	0.26	0.28	0.30	0.31	0.28

retrieval performance. Our proposed model SCANet is based on GCN text features and VGG-19 [17] image features. We first fix the image features of VGG-19 and change the text features by LSTM and CNN [6], respectively. Then we fix the text features of GCN and change the image features by ResNet-50 [3] and CNN with five convolution layers (CNN-5), respectively. Particularly, CNN-5 is trained end-to-end with our proposed model. Meanwhile, for each baseline model, we conduct four experiments to variate the number of attention layers in the range of (0,1,2,3) and show its affects on the performance. All the experiments are conducted on the Eng-Wiki dataset. The retrieval performance of MAP is given in Table 3.

The Influence of Co-attention Layer Numbers. When varying the number of co-attention layers, we can see that almost all the models benefit from our stacked co-attention method compared with the original models (i.e. #Attention layers=0). Generally speaking, when the number of attention layers changes from 1 to 3, the increase of MAP scores of different models is ranging from 2% to 6%. That’s because the stacked co-attention layers progressively enhance the mutually attended features of the paired text and image and filter out the unaligned noise. Different models fit for different number of co-attention layers and 2 is a relatively good setting in most cases. SCANet with two co-attention layers obtains the highest MAP compared with other baseline models.

The Influence of Text and Image Features. For the first three models in Table 3, we fix the image VGG-19 features. For each number of attention layers, it’s obvious that SCANet outperforms other models especially for the text retrieval task, which indicates the power of GCN in semantic representation of texts. The MAP of LSTM is inferior than GCN while CNN performs the worst. For the last three models in Table 3, we fix the text GCN features. We also obtain the same conclusion that SCANet performs the best on all the number of attention layers. The model using ResNet-50 is slightly worse than using VGG-19. CNN-5 performs the worst because that shallow convolutional networks are detrimental to high-level image feature representation.

The Efficiency of SCANet. The experiments were conducted on a 64-bit Linux machine with 4 Tesla V100-PCIE GPUs each with 16.16 GB memory. The training process on Eng-Wiki costs 1.3 h for GCN+VGG19 model (SCANet), 3.6 h for CNN+VGG19 model and 8.7 h for LSTM+VGG19 model. SCANet costs the least training time since that only 3 parameters need to be learnt for calculating the convolutions within 3 layers of neighborhoods.

Table 4. Experiments on the influence of the parameters m and λ .

m	λ	Text query	Image query	Average
0.50	0.35	0.622	0.463	0.543
0.60	0.35	0.808	0.460	0.634
0.70	0.35	0.643	0.473	0.558
0.60	0.30	0.795	0.450	0.623
0.60	0.40	0.791	0.452	0.621

4.5 Parameter Analysis

We conduct several experiments on the Eng-Wiki datasets to explore how parameters affect the cross-modal retrieval performance. We range the value of m from 0.4 to 0.6 and λ from 0.25 to 0.4 and show some of results in Table 4. We can know that the model is not much sensitive to λ and still performs well when λ is in the interval (0.25, 0.40). On the contrary, m has obvious impact on the retrieval performance. The average MAP scores range from 0.47 to 0.63 when varying the value of λ . So, 0.35 for λ and 0.6 for m are the relative best settings.

5 Conclusion

In this paper, we have proposed a stacked co-attention network to progressively align the semantically relevant features of different modalities and strengthen their fine-grained correlations. In the dual-path end-to-end framework, the attention distribution is jointly learnt with both multimodal feature learning and distance metric learning to benefit each other. Experimental results on three benchmark datasets verify that our model outperforms 11 state-of-the-art methods. Meanwhile, the extensive baseline comparisons indicate that the proposed attention approach can promote cross-modal retrieval performance regardless of the feature representations, though GCN+VGG gains the best performance.

Acknowledgement. This work is supported by the National Key Research and Development Program (Grant No. 2017YFC0820700) and the Fundamental Theory and Cutting Edge Technology Research Program of Institute of Information Engineering, CAS (Grant No. Y7Z0351101)

References

1. Blei, D.M., Jordan, M.I.: Modeling annotated data. In: Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 127–134. ACM (2003)
2. Defferrard, M., Bresson, X., Vandergheynst, P.: Convolutional neural networks on graphs with fast localized spectral filtering. In: NIPS, pp. 3837–3845 (2016)
3. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR, pp. 770–778 (2016)
4. Kan, M., Shan, S., Zhang, H., Lao, S., Chen, X.: Multi-view discriminant analysis. TPAMI **38**(1), 188–194 (2016)
5. Kang, C., Xiang, S., Liao, S., Xu, C., Pan, C.: Learning consistent feature representation for cross-modal multimedia retrieval. TMM **17**(3), 276–288 (2017)
6. Kim, Y.: Convolutional neural networks for sentence classification. arXiv preprint [arXiv:1408.5882](https://arxiv.org/abs/1408.5882) (2014)
7. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv preprint [arXiv:1609.02907](https://arxiv.org/abs/1609.02907) (2016)
8. Kumar, B.G.V., Carneiro, G., Reid, I.: Learning local image descriptors with deep siamese and triplet convolutional networks by minimizing global loss functions. In: CVPR, pp. 5385–5394 (2016)
9. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint [arXiv:1301.3781](https://arxiv.org/abs/1301.3781) (2013)
10. Nam, H., Ha, J.W., Kim, J.: Dual attention networks for multimodal reasoning and matching. arXiv preprint [arXiv:1611.00471](https://arxiv.org/abs/1611.00471) (2016)
11. Peng, Y., Huang, X., Qi, J.: Cross-media shared representation by hierarchical learning with multiple deep networks. In: IJCAI, pp. 3846–3853 (2016)
12. Peng, Y., Qi, J., Yuan, Y.: Modality-specific cross-modal similarity measurement with recurrent attention network. arXiv preprint [arXiv:1708.04776](https://arxiv.org/abs/1708.04776) (2017)
13. Qin, Z., Yu, J., Cong, Y., Wan, T.: Topic correlation model for cross-modal multimedia information retrieval. PAA **19**(4), 1007–1022 (2016)
14. Ranjan, V., Rasiwasia, N., Jawahar, C.: Multi-label cross-modal retrieval. In: ICCV, pp. 4094–4102 (2015)
15. Rasiwasia, N., et al.: A new approach to cross-modal multimedia retrieval. In: ACM-MM, pp. 251–260 (2010)
16. Sharma, A., Kumar, A., Daume, H., Jacobs, D.W.: Generalized multiview analysis: a discriminative latent space. In: CVPR, pp. 2160–2167 (2012)
17. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. Comput. Sci. (2014)
18. Wang, K., He, R., Wang, L., Wang, W., Tan, T.: Joint feature selection and subspace learning for cross-modal retrieval. TPAMI **38**(10), 2010–2023 (2016)
19. Wang, K., He, R., Wang, W., Wang, L., Tan, T.: Learning coupled feature spaces for cross-modal matching. In: ICCV, pp. 2088–2095 (2013)
20. Yang, Z., He, X., Gao, J., Deng, L., Smola, A.: Stacked attention networks for image question answering. In: CVPR, pp. 21–29 (2016)
21. Yu, J., et al.: Modeling text with graph convolutional network for cross-modal information retrieval. arXiv preprint [arXiv:1802.00985](https://arxiv.org/abs/1802.00985) (2018)
22. Zhang, L., Ma, B., He, J., Li, G., Huang, Q., Tian, Q.: Adaptively unified semi-supervised learning for cross-modal retrieval. In: IJCAI, pp. 3406–3412 (2017)

23. Zhang, X., et al.: HashGAN: attention-aware deep adversarial hashing for cross modal retrieval. arXiv preprint [arXiv:1711.09347](https://arxiv.org/abs/1711.09347) (2017)
24. Zhen, Y., Yeung, D.Y.: Co-regularized hashing for multimodal data. In: NIPS, pp. 1376–1384 (2012)
25. Zheng, Z., Zheng, L., Garrett, M., Yang, Y., Shen, Y.D.: Dual-path convolutional image-text embedding. arXiv preprint [arXiv:1711.05535](https://arxiv.org/abs/1711.05535) (2017)



Supervised Manifold-Preserving Graph Reduction for Noisy Data Classification

Zhiqiang Xu and Li Zhang(✉)

School of Computer Science and Technology and Joint International Research Laboratory of Machine Learning and Neuromorphic Computing, Soochow University, Suzhou 215006, Jiangsu, China

20164227004@stu.suda.edu.cn, zhangliml@suda.edu.cn

Abstract. Data reduction has become one of essential techniques in current knowledge discovery scenarios, dominated by noisy data. The manifold-preserving graph reduction (MPGR) algorithm has been proposed, which has the advantages of eliminating the influence of outliers and noisy and simultaneously accelerating the evaluation of predictors learned from manifolds. Based on MPGR, this paper utilizes the label information to guide the construction of graph and presents a supervised MPGR (SMPGR) method for classification tasks. In addition, we construct a similarity matrix using kernel tricks and develop the kernelized version for SMPGR. Empirical experiments on several datasets show the efficiency of the proposed algorithms.

Keywords: Data reduction · Kernel trick · Graph reduction
Manifold learning · Supervised learning

1 Introduction

In real life, the data we encounter often contain noise. Since the quality of models often depends on the quality of given data, noises would have a negative impact on constructing models if left untreated. Therefore, as one technique of data preprocessing, data reduction is particularly important [1,2]. In order to filter noise data and reduce the time of training models, we focus on how to pick out a few representative sample points from the original data.

Recently, there has been considerable research for data reduction problems [3–10]. Data reduction is considered to find potential support vectors for training support vector machines (SVMs), which could efficiently accelerate the training procedure of SVMs [6–10]. Random sampling could be a way to perform data reduction [4]. In addition, a modified K-means algorithm was used to remove noise data. However, these methods cannot guarantee that the manifold structure of data is preserved, and remove the outliers and noisy examples effectively.

To maintain the manifold structure, it is necessary to develop learning methods under the manifold assumption [2,3,11–16]. However, most manifold methods aim at reducing the dimension of data instead of the number of data [16–18],

which will not be discussed here. Fortunately, Sun et al. proposed a manifold-preserving graph reduction (MPGR) method for reducing the amount of data [3], which is able to preserve the manifold structure, and rule out possible outliers and noisy points. In MPGR, some samples are selected according to some criteria to construct a graph, which can maximize the potential popularity structure of the original data. However since MPGR does not take use of label information, the reduced data may be not so clear for classification tasks.

To remedy the issue, this paper proposes a novel manifold-preserving data reduction method inspired by MPGR [3], called supervised manifold-preserving graph reduction (SMPGR). This proposed method uses label information to construct the weight matrix, which can filter the sample points with unreliable label information, and then adopts MPGR to achieve data reduction. Usually, the reduced points often maintain the manifold information. Considering the linearly inseparable case, we present the kernel version for SMPGR (KSMPGR) by introducing the kernel trick in constructing weight matrix. To validate the effectiveness of the proposed methods, we first select data points and then compare the classification performance based on the 1-norm support vector machine [19].

The rest of paper is organized as follows. In Sect. 2, we briefly describe the manifold-preserving graph reduction algorithm, and presents the supervised MPGR methods. Experimental results on synthetic and real datasets are shown in Sect. 3. The last section concludes this paper.

2 Supervised Manifold-Preserving Graph Reduction Method

MPGR aims at getting a sub-graph in an unsupervised way, which may be not the optimal sub-graph for classification tasks. In order to achieve sample selection for classification problems, we propose a supervised manifold-preserving graph reduction algorithm that uses label information as an aid in constructing the weight matrix for data reduction. Thus, the transition from unsupervised graph reduction to supervised preprocessing is achieved.

This section reviews MPGR, discusses how to calculate the weight matrix and proposes SMPGR. In order to solve the problem of linearly inseparability, we give the kernelized SMPGR by introducing kernel functions to construct the weight matrix. At the end, we show the framework of the proposed method and analyze its computational complexity.

2.1 Manifold-Preserving Graph Reduction

MPGR can remove the outliers and noisy examples while preserving the manifold structure of data [3]. Given a graph $G(V, E, \mathbf{W})$ corresponding to a manifold with the vertex set $V = \{\mathbf{x}_1, \dots, \mathbf{x}_m\}$, the number of vertices m , the edge set E and the symmetric weight matrix \mathbf{W} , the goal of MPGR is to get a sub-graph $G_c(V_c, E_c, \mathbf{W}_c)$ with V_c , E_c and \mathbf{W}_c being, respectively, subsets of V , E and \mathbf{W} , where V_c is the retained subset. For the graph adjacency construction, the

Algorithm 1. MPGR

Input: Graph $G(V, E, \mathbf{W})$ with m vertices, and t for the number of the vertices in the desired sub-graph G_c .

Output: Manifold-preserving subgraph G_c with t vertices.

```

1 begin
2   for  $j = 1, \dots, t$  do
3     Compute degree  $d(\mathbf{x}_i)$ ,  $i = 1, \dots, m-j+1$ ;
4     Pick one vertex  $\mathbf{x}$  with the maximum degree;
5     Remove  $\mathbf{x}$  and associated edges from  $G$ ;
6     Add  $\mathbf{x}$  to  $G_c$ ;
7   end
8 end

```

k -nearest-neighbor rule is used where k is the user defined parameter. In MPGR, the weight matrix is defined as:

$$W_{ij} = \begin{cases} \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right), & \text{if } \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ are neighbors} \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

where $\sigma \neq 0$ is the user defined parameter.

Given the number of retained vertices in the sub-graph, the problem of exactly seeking manifold-preserving sub-graphs is NP-hard. MPGR gives a simple and efficient greedy algorithm to construct it. In MPGR, there has an important concept *degree*. Let $d(\mathbf{x}_i)$ be the degree associated with the vertex \mathbf{x}_i , which is defined as

$$d(\mathbf{x}_i) = \sum_{i \sim j} W_{ij} \quad (2)$$

where $i \sim j$ means that vertices \mathbf{x}_i and \mathbf{x}_j are connected by an edge, and the weight W_{ij} between \mathbf{x}_i and \mathbf{x}_j is the i th row and the j th column of \mathbf{W} . Note that if \mathbf{x}_i and \mathbf{x}_j are not linked, then $W_{ij} = 0$.

2.2 Construction of Weight Matrix Using of Label Information

MPGR tries to maintain the manifold of the original datasets using few points without considering label information. For those noisy points that do not affect the structure but also do not have reliable labels, MPGR cannot exclude these points. Thus, the reduced data may be not fit for classification tasks. To solve this issue, we present the supervised MPGR method by introducing label information.

Consider a two-class classification problem. Let $X = \{(\mathbf{x}_i, y_i)\}_{i=1}^m$ be the set of two-class samples, where samples $\mathbf{x}_i \in X \subseteq \mathbb{R}^d$, X denotes the sample space, d is the dimension of samples and labels $y_i \in \{\pm 1\}$. Similar to MPGR, we also use the k -nearest-neighbor rule to construct the weight matrix. For the vertex \mathbf{x}_i , we can get k labels $\hat{y}_1, \dots, \hat{y}_k$ corresponding to its nearest neighbors.

We define the quality of neighbors for \mathbf{x}_i by

$$q_i = \sum_{j=1}^k y_i \hat{y}_j \quad (3)$$

where y_i is the label of \mathbf{x}_i . We have a new way to compute the weight matrix and rewrite (1) as:

$$W_{ij} = \begin{cases} \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right), & \text{if } \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ are neighbors \& } q_i > 0 \\ 0, & \text{otherwise.} \end{cases} \quad (4)$$

It is noteworthy that the sample \mathbf{x}_i has a negative q_i if and only if half of its neighbors belong to the other class. Thus, it is reasonable to take \mathbf{x}_i as a noisy point. In this case, the weights between \mathbf{x}_i and its neighbors are zero.

From Fig. 1, we can see those noisy points that do not affect the structure and do not have reliable labels. We illustrate the ability of removing noisy points of SMPGR, as shown in Fig. 2. MPGR cannot exclude those noisy points, just as given in Fig. 2(a). The reduced result obtained by SMPGR is shown in Fig. 2(b), where noisy points are deleted.

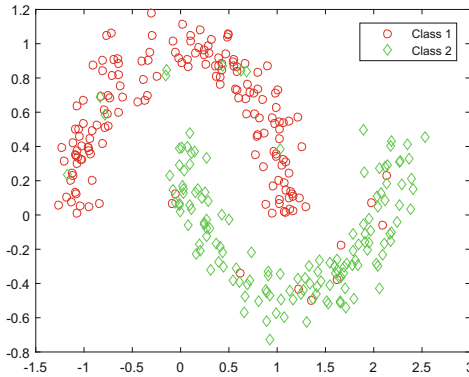


Fig. 1. Original synthetic dataset

2.3 Kernelized Version

In real world, the problems encountered cannot always be handled using linear methods. In order to make the method able to accommodate nonlinear cases, we extend it to the nonlinear counterpart by well-known kernel tricks [20, 21]. First, we define a set of nonlinear mappings [22, 23]:

$$\begin{aligned} \Phi: \mathbf{x}_i \in X \subseteq \mathbb{R}^d &\xrightarrow{\Phi} \\ \Phi(\mathbf{x}_i) &= [\phi_1(\mathbf{x}_i), \phi_2(\mathbf{x}_i), \dots, \phi_D(\mathbf{x}_i)] \in F \subseteq \mathbb{R}^D \end{aligned} \quad (5)$$

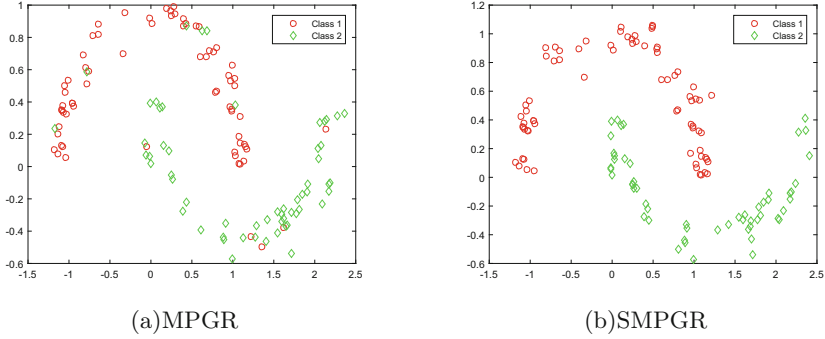


Fig. 2. An illustration, where retained data obtained by MPGR (a) and SMPGR (b).

where F denotes the feature space with the dimension D , $\Phi(\mathbf{x}_i)$ is the image of \mathbf{x}_i in the feature space, each mapping function $\phi_j(\mathbf{x}_i)$ may take any nonlinear function [22–24]. Accordingly, the weight matrix in the feature space is defined as follows.

$$W_{ij} = \begin{cases} \exp\left(-\frac{\|\Phi(\mathbf{x}_i) - \Phi(\mathbf{x}_j)\|^2}{2\sigma^2}\right), & \text{if } \Phi(\mathbf{x}_i) \text{ and } \Phi(\mathbf{x}_j) \text{ are neighbors \& } q_i > 0 \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

where $\|\Phi(\mathbf{x}_i) - \Phi(\mathbf{x}_j)\|^2$ can be expanded into the form of

$$\langle \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_i) \rangle + \langle \Phi(\mathbf{x}_j) \cdot \Phi(\mathbf{x}_j) \rangle - 2\langle \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j) \rangle.$$

Typically, a Mercer kernel function can be represented as

$$k(\mathbf{x}_i, \mathbf{x}_j) = \langle \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j) \rangle$$

Therefore, in the feature space we have

$$W_{ij} = \begin{cases} \exp\left(-\frac{d^2(\Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j))}{2\sigma^2}\right), & \text{if } \Phi(\mathbf{x}_i) \text{ and } \Phi(\mathbf{x}_j) \text{ are neighbors \& } q_i > 0 \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

where $d^2(\Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j)) = k(\mathbf{x}_i, \mathbf{x}_i) + k(\mathbf{x}_j, \mathbf{x}_j) - 2k(\mathbf{x}_i, \mathbf{x}_j)$.

2.4 Algorithm Description

Once the weight matrix is determined, the way of selecting points in MPGR can be applied to the proposed methods. Algorithm 2 describes the proposed methods SMPGR and KSMPGR, where we also seek t vertices from the original graph of m vertices. SMPGR or KSMPGR first constructs a weight matrix according to (4) or (7), then selects the sample with the maximum degree (if more than one sample has the same maximum degree, we randomly pick one), and removes the corresponding row and column in the matrix \mathbf{W} . This step tends to select

Algorithm 2. SMPGR/KSMPGR

Input: The set of two-class samples X and the number of the samples we desire t .

Output: Target sample set X_t .

```

1 begin
2   Compute  $\mathbf{W}$  by (4) or (7);
3   Initialize the target sample set  $X_t = \emptyset$ ;
4   for  $j = 1, \dots, t$  do
5     Compute degree  $d(\mathbf{x}_i)$ ,  $i = 1, \dots, m-j+1$ ;
6     Pick one sample  $\mathbf{x}^*$  with the maximum degree, namely
           
$$\mathbf{x}^* = \operatorname{argmax}_{\mathbf{x}_i \in X} d(\mathbf{x}_i);$$

7     Remove  $\mathbf{x}^*$  from  $X$  and the corresponding row and column from the
           matrix  $\mathbf{W}$ ;
8     Move  $\mathbf{x}^*$  into  $X_t$ ;
9   end
10 end

```

successive samples with a high space connectivity. Then, it adds the selected samples in a target sample set (which is null initially). The same procedure repeats on the resultant weight matrix until the target sample set has t samples.

There are two main steps in both SMPGR or KSMPGR, construction of \mathbf{W} and selecting manifold-preserving points. Suppose that t vertices are sought from an original graph with m vertices, and the dimension of sample is d . Let n_g be the maximum number of edges linked to a vertex in the original graph. The complexity of constructing \mathbf{W} in both SMPGR and KSMPGR is $O(dm^2)$. The complexity of selecting manifold-preserving points is $O(n_g mt)$. Thus, the total computational complexity of our algorithm is $O(dm^2 + d_g mt)$. The proposed methods have the same computational complexity as MPGR with considering the construction of \mathbf{W} .

3 Experiments

In this section, we demonstrate the effectiveness of the proposed methods on different datasets, including noisy synthetic data, face images and handwritten digits. The goal of our methods is to select good training samples which are useful for the construction of classification models. We take 1-norm SVMs and kernel 1-norm SVMs [19] as the subsequent classifiers to compare the validity of data reduction methods, including random sampling, MPGR, SMPGR and KSMPGR.

Let the nearest number of neighbors $k = 10$ when constructing the weight matrix, which follows the experimental setting in [3]. For KSMPGR, we adopt the radial basis function (RBF) kernel $k(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2/2\sigma^2)$ with the kernel parameter σ , which is a part of (1) or (4). For MPGR, SMPGR, and

KSMPGR, the parameter σ is determined by using the median method [28], or taking the median value in the set $\{\|\mathbf{x}_i - \bar{\mathbf{x}}\|^2\}_{i=1}^m$ where $\bar{\mathbf{x}}$ is the mean of all training samples. The code of our algorithms is written in MATLAB R2013a on a PC with an Inter Core I5 processor with 4GB RAM. In this paper, each experiment is repeated 10 times.

3.1 Noisy Synthetic Data

We generate a two-moon dataset, which has 1000 training examples and 1000 test ones for each class [3]. Here, we validate the effect of the noisy level and the retained sample number on performance of compared methods. We then observe the classification accuracy of 1-norm SVMs and kernel 1-norm SVMs based on the sample points selected by different data preprocessing methods.

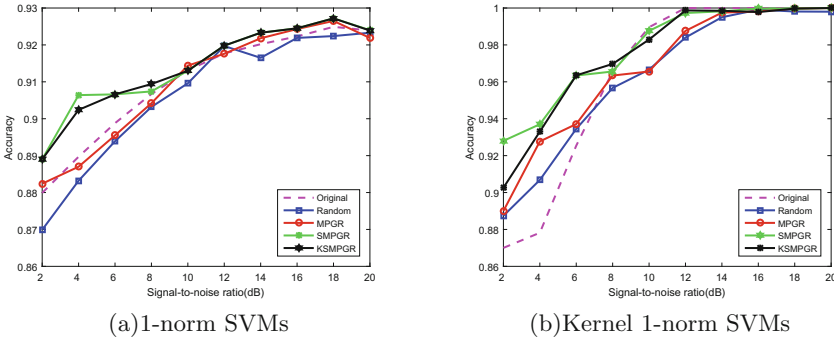


Fig. 3. Classification accuracy vs. SNR using different classifiers, (a) 1-norm SVMs, and (b) kernel 1-norm SVMs on synthetic data.

(1) Experiments on noisy level

First, we fix the number of retained points t and then vary signal-to-noise ratio (SNR) in the original dataset. Let t be 10% of the total training samples, or $t = 0.1m$ with the number of training samples m . Each training sample is corrupted by the Gaussian white noise, whose level varies from 2 dB to 20 dB. The average results on 10 trails are reported in Fig. 3.

Figure 3 shows the curves of classification accuracy vs. SNR using 1-norm SVMs and kernel 1-norm SVMs, respectively, where “None” means that none of data reduction method is used, and “Random” means the random sampling method.

When taking the 1-norm SVMs as the classifier, we can see that both SMPGR and KSMPGR outperform MPGR and random sampling if SNR is small, which means that the proposed methods are robust when the noisy level is large. Obviously, random sampling is the worst among these methods. The 1-norm SVMs

do not have the ability removing noisy points, its performance (see “None”) is inferior to SMPGR and KSMPPGR.

The kernel 1-norm SVMs have the ability of removing possible outliers and noisy points due to its sparseness on the decision model. The kernel 1-norm SVMs perform well when the noise level is relative small, say 12 dB, see Fig. 3(b). However, when the noise level is high, the kernel 1-norm SVMs work bad. Thus, we need the data reduction methods to improve its performance. Among these reduction methods, our proposed methods have a better performance when SNR is less than 8 dB.

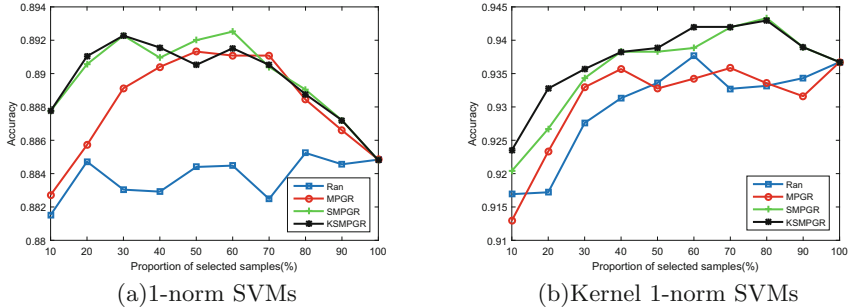


Fig. 4. Classification accuracy vs. proportion of retained samples using different classifiers, (a) 1-norm SVMs, and (b) kernel 1-norm SVMs on synthetic data.

In a nutshell, both SMPGR and KSMPPGR are robust when the noisy level is large, compared to MPGR and random sampling.

(2) Experiments on retained sample number

Second, we corrupt the dataset with a fixed Gaussian white noise of 6 dB and check the effect of the retained sample number on classification performance. Since the number of retained samples needs to be manually assigned, we set $t = rm$, and $r \in \{0.1, 0.2, \dots, 1\}$.

The average experimental results are given in Fig. 4. Basically, both SMPGR and KSMPPGR achieve the best performance. In Fig. 4(a), we can see that the best accuracy with data reduction is about 89.3% when 1-norm SVMs are taken as the classifier. As the number of retained sample increases, the classification performance is at first increasing and then decreasing. The performance of MPGR, for example, decreases from $r = 0.7$.

From Fig. 4(b), we may need more retained samples to achieve the best performance when using the nonlinear classifier compared to the linear one. For example, our methods need 80% retained samples, and MPGR requires 100%. In other words, data reduction may be not so significant when applying nonlinear classifiers.

Totally, we get an insight from Fig. 4. For linear classifiers, we can select a small proportion of the whole data, which can make our methods train a best

model. Contrarily, nonlinear classifiers need a large proportion to achieve better performance.

3.2 MNIST Dataset

This MNIST dataset comes from the UCI Machine Learning Repository [27], which has a total of 60000 training and 10000 test images with total 10 classes. Each image contains 28×28 pixels. Consider digits 3 and 8. The subset used here includes 11982 training examples and 1984 testing examples. We randomly select 10% samples from the training set for training classifiers.

The classification results of different algorithms are given in Fig. 5. Basically, both SMPGR and K SMPGR can achieve the best performance. In Fig. 5(a), we can see that the best accuracy with data reduction is about 95.5% when 1-norm SVMs are taken as the classifier. From Fig. 5(b), we can also see that best performance is 96.5%.

Totally, we get an insight from Fig. 5. For linear classifiers, we can select 70% of the whole data, which can make our methods train a best model. Contrarily, nonlinear classifiers need 80% to achieve better performance.

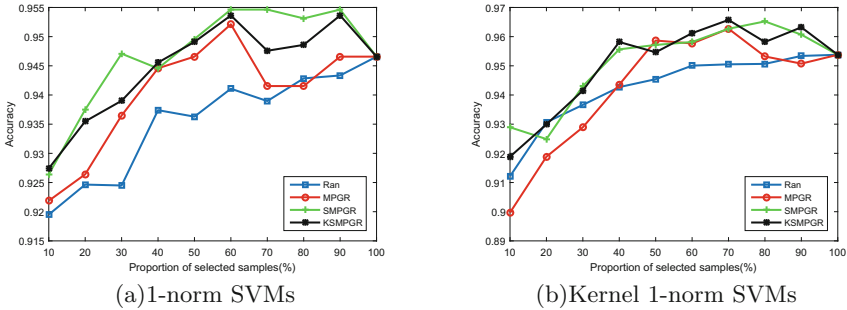


Fig. 5. Classification accuracy vs. proportion of retained samples using different classifiers, (a) 1-norm SVMs, and (b) kernel 1-norm SVMs on MNIST.

3.3 CBCL Face Dataset

Face detection is a binary classification problem which intends to identify whether a picture is a human face or not. In this experiment, 2429 face and 4548 non-face images from the MIT CBCL repository are used [25,26], where half of them are faces and each image is a 19×19 gray picture, we choose 10% of images as the training set and the rest as the test set. The same experimental setting with the previous handwritten digit classification is adopted.

The classification results of different algorithms are given in Fig. 6. We have the same conclusion as derived from previous experiments. Both SMPGR and K SMPGR are much better than the other two methods in despite of used classifiers.

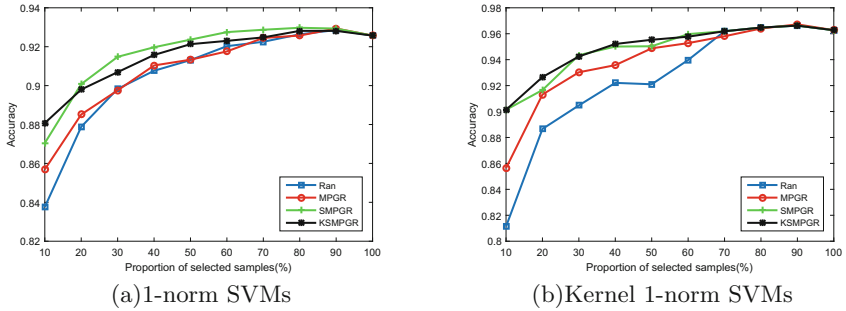


Fig. 6. Classification accuracy vs. proportion of retained samples using different classifiers, (a) 1-norm SVMs, and (b) kernel 1-norm SVMs on CBCL.

4 Conclusion

In this paper, we propose supervised manifold-preserving graph reduction methods for data reduction. The proposed methods can preserve the manifold structure of data using the label information, and rule out possible outliers and noisy points. Experimental results on the noisy synthetic data set indicate that the proposed methods are more robust than MPGR. On the real-world datasets, classifiers with data reduction can obtain better performance than those without data reduction. Among the compared data reduction methods, the proposed methods outperform than others. Especially when training data include outliers and noisy examples, SMPGR/KSMPGR can effectively remove them and even improve classification performance.

Acknowledgment. This work was supported in part by the National Natural Science Foundation of China under Grants No. 61373093, No. 61402310, No. 61672364 and No. 61672365, by the Soochow Scholar Project of Soochow University, by the Six Talent Peak Project of Jiangsu Province of China.

References

1. Pyle, D.: Data preparation for data mining. *Appl. Artif. Intell.* **17**(5–6), 375–381 (1999)
2. Belkin, M., Niyogi, P.: Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Comput.* **15**(6), 1373–1396 (2003)
3. Sun, S., Hussain, Z., Shawe-Taylor, J.: Manifold-preserving graph reduction for sparse semi-supervised learning. *Neurocomputing* **124**(2), 13–21 (2014)
4. Madigan, D., Nason, M.: Data reduction: sampling. In: *Handbook of Data Mining and Knowledge Discovery*, pp. 205–208 (2002)
5. Barca, J.C., Rumantir, G.: A modified K-means algorithm for noise reduction in optical motion capture data. In: *6th IEEE/ACIS International Conference on Computer and Information Science in Conjunction with 1st IEEE/ACIS International Workshop on e-Activity*, pp. 118–122 (2007)

6. Ou, Y.Y., Chen, C.Y., Hwang, S.C., Oyang, Y.J.: Expediting model selection for support vector machines based on data reduction. *IEEE Int. Conf. Syst.* **1**, 786–791 (2003)
7. Burges, C.J.C.: Geometry and invariance in kernel based methods. In: *Advances in Kernel Methods* (2008)
8. Panda, N., Chang, E.Y., Wu, G.: Concept boundary detection for speeding up SVMs. In: *23rd International Conference on Machine Learning*, pp. 681–688 (2006)
9. Jinlong, A.N., Wang, Z.: Pre-extracting support vectors for support vector machine. In: *5th International Conference on Signal Processing*, vol. 3, pp. 1432–1435 (2000)
10. Zhang, L., Zhou, W., Chen, G., Zhou, H., Ye, N., Jiao, L.: Pre-extracting boundary vectors for support vector machine using pseudo-density estimation method. In: *International Symposium on Multispectral Image Processing and Pattern Recognition*, vol. 7496, pp. 74960J–74960J-7 (2009)
11. Rowels, S.T., Saul, L.K.: Nonlinear dimensionality reduction by locally linear embedding. *Science* **290**(5500), 2323–2326 (2000)
12. Sindhwani, V., Rosenberg, D.S.: An RKHS for multi-view learning and manifold co-regularization. *Int. Conf. Mach. Learn.* **307**, 976–983 (2008)
13. Tenenbaum, J.B., De, S.V., Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction. *Science* **290**(5500), 2319–2323 (2000)
14. He, X., Cai, D., Yan, S., Zhang, H.J.: Neighborhood preserving embedding. *Int. Conf. Comput. Vis.* **2**, 1208–1213 (2005)
15. Hinton, G., Roweis, S.: Stochastic neighbor embedding. *Adv. Neural Inf. Process. Syst.* **41**(4), 833–840 (2002)
16. Sammon, J.W.: A nonlinear mapping for data structure analysis. *IEEE Trans. Comput.* **c-18**(5), 401–409 (2006)
17. Hinton, G., Roweis, S.: Stochastic neighbor embedding. *Adv. Neural Inf. Process. Syst.* **41**(4), 833–840 (2002)
18. Shaw, B., Jebara, T.: Structure preserving embedding. *Int. Conf. Mach. Learn.* **382**, 937–944 (2009)
19. Zhang, L., Zhou, W.: On the sparseness of 1-norm support vector machines. *Neural Netw.* **23**(3), 373–385 (2010)
20. Kivinen, J., Smola, A.J., Williamson, R.C.: *Learning with Kernels*. MIT Press, Cambridge (2002)
21. Zhang, H., Huang, W., Huang, Z., Zhang, B.: A kernel autoassociator approach to pattern classification. *IEEE Trans. Syst. Man Cybern.* **35**(3), 593–606 (2005)
22. Zhou, W., Zhang, L., Jiao, L.: Hidden space principal component analysis. In: Ng, W.-K., Kitsuregawa, M., Li, J., Chang, K. (eds.) *PAKDD 2006*. LNCS (LNAI), vol. 3918, pp. 801–805. Springer, Heidelberg (2006). https://doi.org/10.1007/11731139_93
23. Zhang, L., Zhou, W., Jiao, C.: Hidden space support vector machines. *IEEE Trans. Neural Netw.* **15**(6), 1424–1434 (2004)
24. Han, M., Yin, J.: The hidden neurons selection of the wavelet networks using support vector machines and ridge regression. *Neurocomputing* **72**(1–3), 471–479 (2008)
25. Alvira, M., Rifkin, R.: An empirical comparison of SNoW and SVMs for face detection. Massachusetts Institute of Technology (2001)
26. Sun, S.: Ensembles of feature subspaces for object detection. In: Yu, W., He, H., Zhang, N. (eds.) *ISNN 2009*. LNCS, vol. 5552, pp. 996–1004. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-01510-6_113

27. UCI machine learning repository. <http://archive.ics.uci.edu/ml/datasets.html>. Accessed 21 Mar 2018
28. Zhang, L., Zhou, W., Chang, P., Liu, J., Yang, Z., Wang, T.: Kernel sparse representation-based classifier. *IEEE Trans. Signal Process.* **60**(4), 1684–1695 (2012)



Personalize Review Selection Using PeRView

Muhammad Al-khiza'ay¹(✉), Noora Alallaq¹, Qusay Alanoz², Adil Al-Azzawi^{3,4},
and N. Maheswari⁵

¹ School of Information Technology, Faculty of Science, Engineering and Built Environment, Deakin University, Geelong, Australia
{malkhiza,nalallaq}@deakin.edu.au

² Faculty of Mathematics and Computer Science, University of Lodz, Lodz, Poland
qusay.arif2015@gmail.com

³ Electrical Engineering and Computer Science Department,
University of Missouri-Columbia Missouri, Columbia, USA
aaadn5@mail.missouri.edu

⁴ College of Science, Computer Science Department, University of Diyala,
Baqubah, Iraq

⁵ School of Computing Science and Engineering, Vellore Institute of Technology,
Chennai, India
maheswari.n@vit.ac.in

Abstract. In the contemporary era, online reviews have an impact on people of all walks of life while choosing appropriate reviews that satisfied user preferences. Personalized reviews selection that is highly relevant to high coverage concerning matching with micro-reviews is the main problem that is considered in this paper. Toward this end, select a personalized subset of reviews are suggested. However, none of the existing research has taken into consideration the personalization of reviews. We proposed a framework known as *PeRView* for personalized review selection using micro-reviews. The proposed approach shows that our framework can determine and select the best subset of personalized reviews. Based on metric evaluation approach which considered personalized matching score and subset size.

Keywords: Online reviews · Micro-reviews · Review selection
Personalized review selection

1 Introduction

Online reviews for almost any product or service start influencing the customers' decisions today. A review basically produces an assessment about a company, service, or any other task that under performance Vasconcelos et al. [16]. In these day, we can find a Variety of ample reviews in such different web sources. For instance, in online shopping web services such as (<http://Amazon.com>) which

provides hosts reviews as a part of the online shopping experience. Intuitively, customers are more inundated by the variety of comments that made during numerous reviews. In this case, it is not clear enough that which reviews are worth or not for the customers' attention which is worsened by two factors, length, and verbosity of many reviews. Many contents of the reviews may not content wholly relevant to the main product or service that under reviewing. Moreover, it is more increasingly that is more difficult to discover and determine the authenticity of the review which is the additional intensive issue of a review since for those comments that has been written by a customer that sharing his/her experience Bodke et al. [1].

Recently, social networking has been growth intensively and has been become more established from year-to-year as well as the micro-blogging services that has been recently provided. In this case, the emergence of new type of online review has been discovered which called micro-reviews service Nguyen et al. [11]. Micro-review is prevailing since the emerging trends in social media and micro-blogging which is alternative source of information for the reviews to find more interested information to read. Micro-reviews are consisting, short, and focused, as well as they are nicely complementing, elaborate, and verbose nature of full-text reviews focusing on a specific of an item Chong et al. [3]. Moreover, the micro-review cannot be properly expanded more than 140 characters which means that the reviewers should be focusing on aspect of the venue that are more important to the investigators (users) [6].

In this paper, we present rationale and motivating factor approach of selecting the personalized reviews that are highly relevant with high personalized coverage in terms of matching with a selecting sub-set of micro reviews. The selected sub-set contains more focused and concise by reflecting the true opinion of individuals. Moreover, selected sub-set of the micro-reviews are not verbose and do not contain unnecessary information. The idea is to have selection a personalized subset of reviews, pertaining to any restaurant or service, that exhibit high personalized coverage in terms of matching with micro-reviews sub-set. Our matching strategy includes different personalized matching criteria such as syntactical, semantic and sentiment matching. Towards this end, we proposed a framework named PeRView with an underling algorithm named Personalized Review Selection Algorithm (PRSA). Due to limited page number, the full version of the paper on (<https://arxiv.org/abs/1804.08234>).

The rest of the paper is structured as follows. Section 2 related works and preliminary. Section 3 presents the proposed methodology. Section 4 presents experimental results. Section 5 throws light into conclusions drawn.

2 Related Works and Preliminary

2.1 Related Works

This section reviews important literature related to online review selection. [14] employed selection concept with data mining concepts but it was meant for personnel selection. Lappas et al. [7] focused on review selection. Their work

was meant for filling the gap between review summarization and review selection processes. Tsaparas et al. [15] studied user reviews made online and explored a method for selecting a comprehensive collection of reviews that make sense.

Ganesan et al. [5] made it explored an unsupervised approach for generating summary of opinions. They proposed a methodology for generating ultra-concise summaries of sentiments. Nguyen et al. [10] also used micro-reviews in order to have efficient selection of reviews. They used micro-reviews to obtain salient features of reviews and finally select best reviews.

Nguyen et al. [12], they explored review synthesis for summarization of micro-reviews for making them more compact and readable. Nguyen et al. [11] studied mining of massive textual data in order to obtain heuristics for achieving selection problem. More on feature selection algorithms is found in [2,9,18]. Niu et al. [13] used short comments and investigated the problem of rated aspect summarization of target entity to have knowledge on such data. Wang et al. [17] studied the users of online review, they tried to estimate different aspects of users personality. Many researchers contributed towards analysis of online reviews [4,8,11]. In this paper we improve review selection used micro-reviews to find personalization.

2.2 Preliminary

Literary, review and micro-review comments are a set of words where the single word mathematically denoted as W . Although, the review is a group of sentences that are part of a review denoted as the following Eq. 1:

$$R_{R \in \mathbb{R}} = \{s_1, s_2, \dots, s_n\} = \sum_{i=1}^R S_i \quad (1)$$

where s is denoted as a sentence and it is define as a set of words as the following Eq. 2 shows:

$$S_{s \in \mathbb{R}} = \{s_1, s_2, \dots, s_n\} = \sum_{i=1}^S W_i \quad (2)$$

The term Micro-Review which is denoted as MR is also defined as a group of words denoted as the following Eq. 3 shows:

$$MR_{N \in \mathbb{R}} = \{W_1, W_2, \dots, W_n\} = \sum_{i=1}^N W_i \quad (3)$$

where N is the size of the micro-reviews. Moreover, the corpus of reviews in our case which is denoted as $G_{(number\ of\ reviews)}$ is defined as a collection of reviews and micro-reviews that are available in the dataset which is mainly denoted as the following Eq. 4 shows:

$$G_{C \in \mathbb{R}} = \{D_1, D_2, \dots, D_n\} = \sum_{i=1}^C D_i = \sum_{i=1}^C \left(\sum_{j=1}^{C_R} R_{C_R} + \sum_{k=1}^{C_{MR}} MR_{C_{MR}} \right) \quad (4)$$

Where C_R and C_{MR} are the number of the reviews and micro-reviews separately. The personalization term also refers to the fact that the review selection is associated with a user and his/her preferences. Moreover, the term Preferences which is denoted as P is defined as a set of likings of user denoted as the following Eq. 5 shows:

$$P_{P \in \mathbb{R}} = \{P_1, P_2, \dots, P_n\} = \sum_{i=1}^P P_i \quad (5)$$

where $P_i (1 \leq i \leq n)$.

The term *Preferences of Previous Users* which is denoted as pp , is defined as set of preferences of previous users denoted as the following Eq. 6 shows.

$$PP_{PP \in \mathbb{R}} = \{PP_1, PP_2, \dots, PP_n\} = \sum_{i=1}^P P_i \quad (6)$$

where $PP_i (1 \leq i \leq n)$.

Another term is the Matching Function which is denoted as F , is defined as the following Eq. 7 shows:

$$F_{Score}(r, mr) = \sum_{i=1}^R \sum_j^{MR} i^{min}(r_i, mr_j) \quad (7)$$

where s is a sentence in R and MR are a reviews and micro reviews. The function also considers P and PP while checking similarity. Selection Coverage which is another term in our problem that is denoted as $Coverge(R)$ and it is defined as the maximum number of micro-reviews matching with limited number of reviews that satisfy user preferences as it shown in Eq. 8:

$$Coverge(R) = \max_{MR} \left((|R \subseteq MR|), \left(\frac{P}{|MR|} \right) \right) \quad (8)$$

The main criteria of the selection majority in this problem is the *Selection Efficiency* which refers to the efficiency of a review R . In another word, the *Selection Efficiency* is nothing but fraction of relevant sentences in R that satisfy user preferences as the following Eq. 9 shows:

$$Efficiency(R) = \frac{|R^r|}{|R|} \quad (9)$$

3 Proposed Methodology

The proposed system aims to facilitate the mining of reviews and micro-reviews in order to discover personalized reviews that satisfy the user preferences. It provides various components that can be used to complete the research and evaluate the work.

3.1 Profile Builder

Our approach for the profile builder has three main stages. Firstly, is the pre-processing stage where the input for this stage is unstructured text data from the reviews dataset that include stop word and stemming process. Secondly, is Keywords extraction and dictionary builder which consists the high frequently words after applying the Histogram of Words (WoF). Finally, is the profile builder by using TF/IDF matrices creator. The TF/IDF which is a standard measure to reflect the importance of a word to a document with respect to the corpus. Based on the return results of the TF/IDF each document is assigned to one category to build the profile for this document.

3.2 Sub Micro-review Set Selection

In this section, we will select a small set of micro-reviews that cover as many reviews as possible, with few sentences. Mathematically, we call the micro-review $MR_{N \in \mathbb{R}}$ efficiency covers the whole Review set $R_{R \in \mathbb{R}}$ if a Micro-Review sentence $S_{S \in MR_S}$ matches the review comment. In another word, micro-review $MR_{N \in \mathbb{R}}$ covers any review $R_{R \in \mathbb{R}}$ if there is a sentence $S_{S \in MR} \in MR_{N \in \mathbb{R}}$. Micro-reviews sub-set is denoting that $T_{N \subseteq MR}$ is a sub-set of Micro-Reviews $t_{MR \in \mathbb{R}}$ that the Reviews set T_R is covered by at least one sentence from the micro-review $MR_{N \in \mathbb{R}}$ sentences. Formally, we can define the whole problem as Eq. 10 shows below:

$$T_R = t_{MR} \in T_{MR_{t_{MR}}} : \exists S_{S \in t_{MR}} \in MR_{N \in \mathbb{R}}, R_{R \in \mathbb{R}}, F(S, R) = 1 \quad (10)$$

we say that T_{MR} sub-set covers the review topic T_R by defining the coverage of review R as the following Eq. 11.

$$Cov(MR) = \frac{T_{MR}}{T_R} \quad (11)$$

We can extend this definition to the case of a collection of a subset of micro-reviews. For a set of micro-reviews $S \subseteq MR$, we define the coverage of the set S as Eq. 12 shows below:

$$Cov(S) = \frac{|\cup_{MR \in S} T_{MR}|}{T_R} \quad (12)$$

To achieve the efficiency selection criteria or a micro-review set $MR_{N \in \mathbb{R}}$ let assume that MR^{mr} of such "relevant" sentences which cover at least one review topic as show in 13:

$$MR^{mr} = S_{s \in MR} \in MR_{N \in \mathbb{R}} : \exists t_{MR} \in T_{MR}, F(S, t) = 1 \quad (13)$$

Then, the define the efficiency $efficiency(MR)$ is define as a fraction of "relevant" sentences in MR. Which is formally can be written as the Eq. 14 shows:

$$Eff(MR) = \frac{|MR^{mr}|}{|MR|} \quad (14)$$

Hence, use the average efficiency of a set S is defined as the average efficiency of the micro-reviews in the set. Formally, we can define that as the following Eq. 15:

$$Eff_{Average(S)} = \frac{\sum_{MR \in S} Eff(S)}{|S|} \quad (15)$$

The algorithm, shown in Algorithm 1, proceeds in iterations each time adding one review to the collection S . At each iteration, for each review MR we compute two quantities. The first is the gain $gain(MR)$, which is the increase in coverage that we obtain by adding this micro-review to the existing collection S . The second quantity is the cost $Cost(R)$ of the review MR , which is proportional to the inefficiency $1 - Eff(R)$ of the review, that is, the fraction of sentences of MR that are not matched to any review. We select the micro-review MR^* that has the highest gain-to cost ratio and guarantees that the efficiency of the resulting collection is at least α , where α is a parameter provided in the input.

Algorithm 1. Micro-Reviews Selection Algorithm

Input: Set of Micro-reviews MR , Set of Reviews R , Efficiency function Eff ; Threshold T : selection number of the micro-reviews, parameters α, β

Output: Set of Micro-reviews $S \subseteq MR$ of size T .

```

1:  $S=0$ ;
2: if  $|S| < T$  then
3:   for all  $MR \in R$  do
4:      $gain(MR) = Cov(S \cup MR) - Cov(S)$ 
5:      $Cost(MR) = \beta(1 - Eff(MR)) + (1 - \beta)$ 
6:   end for
7:    $\varepsilon = \max_{MR \in R : Eff(S \cup MR) \geq \alpha}$ 
8:   if ( $\varepsilon == 0$ ) or  $\max_{MR \in \varepsilon} gain(MR) == 0$  then
9:     Break
10:  end if
11:   $MR^* = \arg\max_{MR \in \varepsilon} gain(MR) / cost(MR)$ 
12:   $S = S \cup MR^*$ 
13:   $R = R / MR^*$ 
14: end if
15: return  $S$ 

```

3.3 Our Proposed Framework Personalized Reviews Selector (PeRView)

Our proposed framework is called PeRView is illustrated in Fig. 4. The proposed Algorithm 2 takes set of reviews of chosen restaurant, set of micro-reviews of same restaurant, user preferences for personalization and matching threshold as input and produces set of reviews that exhibited high quality and coverage. For each review all micro-reviews are compared for similarity. Similarity is found syntactically, semantically and polarities related to sentiments. Then the similarities are merged to have final quantitative value.

Algorithm 2. Personalized Review Selection Algorithm

Input: Set of reviews \mathbf{R} , set of micro-review \mathbf{MR} related to \mathbf{R} , user preferences \mathbf{P} , threshold t **Output:** Set of reviews that with high quality and coverage \mathbf{R}'

```

1: initialize  $\mathbf{R}'$  to hold result
2: for each review  $\mathbf{r}$  in  $\mathbf{R}$  do
3:   for each sentence  $\mathbf{s}$  in  $\mathbf{r}$  do
4:      $count \leftarrow 0$ 
5:     for each micro review  $\mathbf{mr}$  in  $\mathbf{MR}$  do
6:        $sim_1 \leftarrow find - Syntactic - Sim(s, mr)$ 
7:        $sim_2 \leftarrow find - Semantic - Sim(s, mr)$ 
8:        $sim_3 \leftarrow find - Sentiment(s, mr)$ 
9:        $sim \leftarrow Rankink - evaluation(sim_1, sim_2, sim_3)$ 
10:       $sim \leftarrow Selection(sim_1, sim_2, sim_3)$ 
11:      if  $\mathbf{s}$  covers  $\mathbf{mr}$  and  $\mathbf{P}$  with  $\mathbf{sim}$  then
12:        increment  $\rightarrow count$ 
13:      end if
14:    end for
15:  end for
16:  if  $count \geq t$  then
17:    add review  $\rightarrow \mathbf{R}'$ 
18:  end if
19: end for
20: return  $\mathbf{R}'$ 

```

4 Experimental Results

4.1 Dataset

Dataset corpus is collected from two sources. They are (www.YELP.COM) and (www.Foursquare.com). Reviews on restaurants are collected from Yelp while the micro-reviews related to the same entities are collected from Foursquare.

4.2 Evaluation Metric

There are two main factors that we have include in our elevation and selection formula. Personalized matching score and reviews set size. Our evaluation problem definition differs depending on the choice the minim set that has the higher personalized coverage score. Hence, we define the efficiency personalized scoring function $PerEff_{min}$ which is defined as the minimum set of selected reviews \mathbf{S} that has a higher personalized coverage score. Formally, we define the minimum efficiency selected set by:

$$PerEff_{min} = \min_{R \in S}(R) \quad (16)$$

The *Personalized – Eff_{min}* of each individual Personal selected review must have a personalized similarity score that by computing the personalized efficiency score should be at least α as a constraint which presents the personalized coverage evaluation score. Formally we define the that as:

$$PerEff_{min}(S) \geq \alpha \quad (17)$$

The size of the personalized selected reviews set is the second factor that our evaluation should consider it. Therefore, we define such an optimization function

called maximization of the personalized coverage score *MaxPerCoverage*, where the reviews personalized similarity scores are restricted to the personal selected review subset size in such should have an efficiency personalized score at least α . In this case the *MaxPerCoverage* optimization function can be used for obtaining an optimal evaluation solution.

Let assume that X_i is obtained as personalized similarity score that associated with each personal review R_i and based on that each individual personal review has been selected. Also, denoting that R_i is the personalized selected set. Although, let assume that Y_i is the sub personalized review size that associate with each personal selected set S_i . We also define another parameter called C_i that associated with each personal subset S_i , with $C_i = 1$ denoting that selected such a personal subset S_i is covered by at least one of review (personalized matching score) in the selected set, and $C_i = 0$ in the otherwise.

To maximize the *MaxPerCoverage* evaluation score such as:

$$\text{maximize } \sum_{j=1}^m C_j = \text{Per} - \text{Simalirity}(s, mr) \tag{18}$$

The *MaxPerCoverage* evaluation score based on some constraint. The first constraint is defined in 19 which ensues the number of selected reviews does not exceed K which means at least one personal review covers the whole set.

$$\text{subject to } \sum_{i=1}^n X_i \leq K \tag{19}$$

The second constraint is defined in 20 which ensures that average personalized similarity matching scores (that covers the whole selected set S_i based on the personalized average score) and also based on the size of its set Y_i to compute the average personalized similarity scoring at least one review that covers C_i must be selected.

$$\sum_{i:S_j \in S, R_i}^n \frac{\sum_{i=1}^n X_i}{Y_i} \geq C_i \forall s_j \in T \tag{20}$$

Finally, the value of both average personalized similarity score and *MaxPerCoverage* evaluation score between the range 0 and 1. Therefore, the main aims of the *MaxPerCoverage* optimization function is to always select the personal review set whose adding maximizes the personalized coverage score and it should be closer than the other reviews based on the approximation personalized ratio for the *MaxPerCoverage*:

$$\text{Approximation Personalized ratio} = \left(1 - \frac{1}{\ell}\right) \tag{21}$$

Where ℓ is the natural logarithm. Hence, we define the *MaxPerCoverage* score is based technically on the *Per-Eff_{min}* efficiency score that has approximation personalized ratio as it defied in 21. Finally, we present the *MaxPerCoverage* algorithm that basically based on many iterations that in each iteration by

adding new review (based on the personalized similarity score) on the collection set S_i based on the personalized matching X_{per_i} score and the set size S_i that is automatically updated upon each new review has been added to the set. In this case, we compute two quantity scores. The first score is the information gain based of the personalized selected review $gain(R_{per})$ which increase the maximization the personalized coverage score based on the average personalized similarity score and the size of the reviews set. Formulary, it is defined in the 22.

$$PerGain(R_{set}) = \left(\frac{\sum_{i=1}^{n=size(y_i)} X_{per_i}}{n} \right) \tag{22}$$

Which in our maximization optimization case 23 will increase the *Maxper-Coverage* score by adding each new closest review (personalized one) to the collected set S_i based on its personalized similarity score.

The second quantity score is the $Cost(R_{per})$ of the personal review \mathbf{R} which is mathematically presents the efficiency personalized score that based on the next formula.

$$PerCost(R_{set}) = 1 - \left(PerEff_{min} = \min_{R_{set} \in S} (R_{set}) \right) \tag{23}$$

Finally, we select the personalized review set that get the highest gain and cost ration based on the personalized matching similarity score and the size of the personalized reviews set. The *MaxPerCoverage* algorithm that is mainly designed to evaluate and select the best personalized review result set is described in Algorithm 3 below:

Algorithm 3. *MaxPerCoverage* Evaluation Algorithm

Input: R_i selected review, x_i personalized matching score, y_i initial reviews set size;

Output: Selected Personal Reviews set $PerC_S$ that has the highest personalized evaluation score $PerC_{R_{set}}$.

- 1: Set the total number of Selected set $T = \sum_{i=1}^n S_i$
 - 2: Set the set size for each Reviews set $Y_i = |S_i|$
 - 3: **for** all S_i **do**
 - 4: Compute the Personalized Evaluation Scores for each set based on
 - 5: $PerC_{R_{set}} = \{gain(R_{set}), Cost(R_{set})\}$
 - 6: Compute the Personalized Gain score for each set based on
 - 7: $PerGain(R_{set}) = \left(\frac{\sum_{i=1}^{n=size(y_i)} X_{per_i}}{n} \right)$
 - 8: Compute the personalized Cost score for each set based on
 - 9: $PerCost(R_{set}) = 1 - (PerEff_{min} = \min_{R_{set} \in S} (R_{set}))$
 - 10: **end for**
 - 11: Get the size of the $PerGain(R_{set})$ scores $C_{R_{set}}$
 - 12: Get the size of the $PerCost(R_{set})$ scores $C_{R_{set}}$
 - 13: Evaluate the set of $PerGain(R_{set})$ and $Cost(R_{set})$
 - 14: **for** $K1 = 1$ to all $PerGain(R_{set})$ **do**
 - 15: **for** $K2 = 1$ to all $PerCost(R_{set})$ **do**
 - 16: **if** $PerGain(R_{set})$ and $PerCost(R_{set})$ is the personalized highest scores, **then**
 - 17: Get the index of the Personalized Reviews set $PerC_{R_{set}} = index(R_{set})$
 - 18: **end if**
 - 19: **end for**
 - 20: **end for**
 - 21: **return** $PerC_{R_{set}}$
-

4.3 Personalized Coverage and Efficiency Selection

In the experimental result we choice different coverage threshold score in case of showing the probability of the personalized matching. Figure 1 shows the different threshold value selection based on the personalized review coverage scoring. Our threshold value selection is significant strong personalized sub-set such as (90%) or (100%) the number of the personalized reviews selection will monotonically decreased. As it is shown in Fig. 3, the number of the personalized reviews selection number is decreased from 15 (reviews) when the threshold value is 50% (personalized coverage-matching score) to 13 (reviews) when the threshold value is increased to be 60%.



Fig. 1. Personalized matching score and achievement accuracy

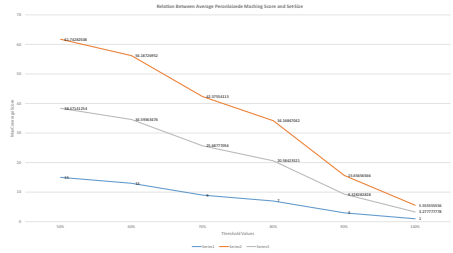


Fig. 2. MaxPerCoverage

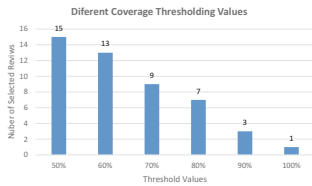


Fig. 3. Personalized matching results

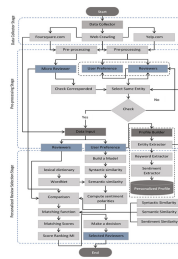


Fig. 4. PeRView framework

The coverage thresholds value is affect the percentage of the achievement score (personalized selection accuracy) as it is shown in Fig. 1, where the best accuracy score for the best personalized reviews selection was in the threshold value (50%) where the total number of the personalized selected reviews is 15 out of dataset reviews which achieve approximately 83.33% accuracy. Then, the achievement accuracy is being decreased among all the other threshold values to reach to (5.55%) in the threshold value (100%) which is significantly tough, but our proposal system is still able to select the best personalized reviews-based on that criteria.

The effectiveness of the *MaxPerCoverage* Evaluation Algorithm shows in Fig. 2. Comparing with the other thresholds values and based on the set-size, two optimal and best personalized review sets will be evaluated. One with the size 15 which achieves 61.74% while the other one with size 13 which achieves 56.18%.

5 Conclusion

In the wake of proliferation of review contents over Internet and their importance in the contemporary world for decision making it is essential to choose high quality personalized reviews that are mainly consist on the main reviewed topic. In this paper we proposed a framework named PeRView which is meant for supporting selection of high quality personalized reviews based on using a sub-set of micro-reviews which consistency more accurate than the reviews based the proposed selection algorithm (*MRS*). In order to find the personalized similarity between a sentence in review and micro review, three kinds of personalized similarity measures are defined and merged. They are known as syntactical similarity which is based on traditional *TF/IDF*, semantic similarity and sentiment similarity based on sentiment polarities. Basically, the evaluation metric is designed based on these personalized similarities as well as the size of the selected reviews to make the final decision for selection the best reviews set. Personalized user preferences are accommodated in the framework to have more qualitative reviews. The experimental results show that the proposed system is able to select the best personalized reviews based on the toughest threshold value (personalized coverage score) which is a 100% accuracy.

Acknowledgement. This work is supported by the practical training project of high-level talents cross-training of Beijing colleges and universities (BUCEA-2018).

References

1. Bodke, A.K., Bhandare, M.G.: Survey on review selection using micro review (2015)
2. Chandrashekar, G., Sahin, F.: A survey on feature selection methods. *Comput. Electr. Eng.* **40**(1), 16–28 (2014)
3. Chong, W.-H., Dai, B.T., Lim, E.-P.: Did you expect your users to say this?: distilling unexpected micro-reviews for venue owners. In: *Proceedings of the 26th ACM Conference on Hypertext and Social Media*, pp. 13–22. ACM (2015)
4. Dai, H., Li, G., Tu, Y.: An empirical study of encoding schemes and search strategies in discovering causal networks. In: Elomaa, T., Mannila, H., Toivonen, H. (eds.) *ECML 2002. LNCS (LNAI)*, vol. 2430, pp. 48–59. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-36755-1_5
5. Ganesan, K., Zhai, C.X., Viegas, E.: Micropinion generation: an unsupervised approach to generating ultra-concise summaries of opinions. In: *WWW*, pp. 869–878. ACM (2012)

6. Kudyba, S.: *Big Data, Mining, and Analytics: Components of Strategic Decision Making*. CRC Press, Boca Raton (2014)
7. Lappas, T., Crovella, M., Terzi, E.: Selecting a characteristic set of reviews. In: *KDD*, pp. 832–840. ACM (2012)
8. Li, Q., Niu, W., Li, G., Cao, Y., Tan, J., Guo, L.: Lingo: linearized grassmannian optimization for nuclear norm minimization. In: *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pp. 801–809. ACM (2015)
9. Mukhopadhyay, A., Maulik, U., Bandyopadhyay, S., Coello, C.A.C.: A survey of multiobjective evolutionary algorithms for data mining: part I. *IEEE Trans. Evol. Comput.* **18**(1), 4–19 (2014)
10. Nguyen, T.-S., Lauw, H.W., Tsaparas, P.: Using micro-reviews to select an efficient set of reviews. In: *CIKM*, pp. 1067–1076. ACM (2013)
11. Nguyen, T.-S., Lauw, H.W., Tsaparas, P.: Review synthesis for micro-review summarization. In: *WSDM*, pp. 169–178. ACM (2015)
12. Nguyen, T.-S., Lauw, H.W., Tsaparas, P.: Micro-review synthesis for multi-entity summarization. *Data Min. Knowl. Discov.* **31**(5), 1189–1217 (2017)
13. Niu, W., et al.: Context-aware service ranking in wireless sensor networks. *J. Netw. Syst. Manag.* **22**(1), 50–74 (2014)
14. Tong, E., et al.: Bloom filter-based workflow management to enable QoS guarantee in wireless sensor networks. *J. Netw. Comput. Appl.* **39**, 38–51 (2014)
15. Tsaparas, P., Ntoulas, A., Terzi, E.: Selecting a comprehensive set of reviews. In: *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 168–176. ACM (2011)
16. Vasconcelos, M., Almeida, J., Gonçalves, M.: What makes your opinion popular?: predicting the popularity of micro-reviews in foursquare. In: *Proceedings of the 29th Annual ACM Symposium on Applied Computing*, pp. 598–603. ACM (2014)
17. Wang, X., Li, G., Jiang, G., Shi, Z.: Semantic trajectory-based event detection and event pattern mining. *Knowl. Inf. Syst.* **37**(2), 305–329 (2013)
18. Xue, B., Zhang, M., Browne, W.N., Yao, X.: A survey on evolutionary computation approaches to feature selection. *IEEE Trans. Evol. Comput.* **20**(4), 606–626 (2016)



An Online GPS Trajectory Data Compression Method Based on Motion State Change

Hui Wang¹, Shuang Liu^{2(✉)}, and Chengcheng Qian³

¹ School of Medical Information, Xuzhou Medical University,
Xuzhou 221004, Jiangsu, People's Republic of China

² Library, Xuzhou Medical University,
Xuzhou 221004, Jiangsu, People's Republic of China
sliu1985@126.com

³ Library, Air Force Logistics College,
Xuzhou 221000, Jiangsu, People's Republic of China

Abstract. Aiming to the problem of insufficient consideration to the cumulative error and offset which online Global Positioning System (GPS) trajectory data compression based on motion state change and the key point insufficient evaluation of online GPS trajectory data compression based on the offset calculation, an online compression of GPS trajectory data based on motion state change named Synchronous Euclidean Distance (SED) Limited Thresholds Algorithm (SLTA) was proposed. This algorithm used steering angle value and speed change value to evaluate information of trajectory point. At the same time, SLTA introduced the SED to limit offset of trajectory point. So SLTA could reach better information retention. The experiment results show that the trajectory compression ratio can reach about 50%. Compared with Thresholds Algorithm (TA), the average SED error of SLTA can be negligible. For other trajectory data compression algorithms, SLTA's average angel error is minimum. SLTA can effectively do online GPS trajectory data compression.

Keywords: GPS trajectory data compression
Synchronous Euclidean Distance thresholds · Motion state change

1 Introduction

As the collection and storage of trajectory data based on temporal and spatial characteristics have shown explosive growth. The GPS trajectory data compression has become one of the focuses of the current research and application. Numerous algorithms about GPS trajectory compression have been proposed. Two contrasting approaches for compressing trajectory data are lossy compression and lossless compression. In contrast with lossless compression, lossy compression can achieve a good compression ratio [1, 2]. Lossy compression has been treated as an active research topic in data compression of the GPS trajectories, amongst which the Douglas-Peucker algorithm [3] is the most popular one. This algorithm divides the line segment with

biggest deviation at each step until the approximated error is smaller than a given error tolerance. After that, Hershberger et al. [4, 5] improved the speed of the DP algorithm for linear simplification. Agarwal et al. [6] improved the DP algorithm for curve simplification. Ma et al. [7] implemented the parallel DP algorithm on multi-processor computers. Keogh et al. [8] proposed the opening window approach (OPW), OPW is no longer iterative for the whole trajectory, but to put forward a “window” concept. OPW starts with a window that contains the first three points of the trajectory and then progressively “opens” the window until a single line segment can no longer represent all of the contained points accurately enough. Then a single line segment from the first point of the window to the second last point in the window is used to approximate the current window, the second last point in the current window becomes the start of the next window. Constantly update the “window” of the information until the completion of the simplified. This method can be synchronized online compression. Muckell et al. [9] proposed spatial quality simplification heuristic (SQUISH) algorithm with buffer concept. They consider a current buffer worth of points and prioritize to keep points within that buffer that are extreme points based on the local estimation of the error. The advantage of this algorithm is that we can set the required compression ratio. Later, they have improved SQUISH algorithm and proposed SQUISH-E [10]. SQUISH-E can not only set the compression ratio, but also set the error threshold. However, the above algorithms neglected the time information of the trajectory. Aiming to this problem, some researchers have proposed improvement methods. Among these researches, Meratnia et al. [11] proposed the top-down time ratio algorithm (TD-TR) and the opening window time ratio algorithm (OPW-TR). They use the SED instead of the vertical distance and consider the time information of the GPS trajectory. They have improved the applicability of the algorithm in GPS trajectory data compression.

However, for online compression of GPS trajectory data about the motion state changes situation, the above algorithms are obviously insufficient for the cumulative error and the offset evaluation. To address these problems, we improve the existing threshold algorithms and propose an online compression of GPS trajectory data based on motion state change named Synchronous Euclidean Distance Limited Thresholds algorithm (SLTA). SLTA uses the steering angle value and speed change value to evaluate information of trajectory point. At the same time, it introduced the SED to limit the offset of trajectory point. SLTA can solve the problem of cumulative error and add the evaluation of offset. It has the ability to evaluate multiple information.

2 Trajectory Data Compression Metrics and Related Concepts

In the initial compression algorithms, most of the GPS trajectory data lossy compression algorithms are derived from ordinary line simplification algorithms. It uses the vertical distance of the general geometric sense to measure the loss of the trajectory information. However, although using the vertical distance can retain the GPS trajec-

tory contour information, the time information of GPS trajectory is almost retained randomly. The introduction of the SED is a solution to solve this problem.

2.1 Synchronous Euclidean Distance

SED is the Euclidean Distance between the point on the original path and this point which corresponds to the time proportion on the simplified path. Figure 1 shows the SED pp' and the vertical distance pq .

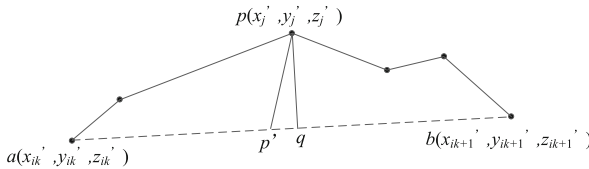


Fig. 1. SED and the vertical distance

For the point p on the original path P , the coordinate of the corresponding point p' on the simplified path P' can be calculated by the following formulas:

$$x'_j = x'_{i_k} + \frac{t'_j - t'_{i_k}}{t'_{i_{k+1}} - t'_{i_k}} \cdot (x'_{i_{k+1}} - x'_{i_k}); \quad t'_{i_k} < t'_j < t'_{i_{k+1}} \quad (1)$$

$$y'_j = y'_{i_k} + \frac{t'_j - t'_{i_k}}{t'_{i_{k+1}} - t'_{i_k}} \cdot (y'_{i_{k+1}} - y'_{i_k}); \quad t'_{i_k} < t'_j < t'_{i_{k+1}} \quad (2)$$

When the position of p' is calculated, the SED between p and p' can be calculated by the following formula:

$$SED(p_i, p'_i) = \sqrt{(x_i - x'_i)^2 + (y_i - y'_i)^2} \quad (3)$$

We can see from the formula (3) that the SED has sufficient consideration for the time information of the trajectory information. SED distributes the moving distance in two adjacent time periods by time and highlights the position of the trajectory point affected by the time. It is more suitable than the vertical distance as the evaluation criterion for the simplifying GPS trajectory data.

2.2 Datum Point of Synchronous Euclidean Distance

The datum point of SED is the first retention point which comes from the simplified trajectory of the predecessor point and the successor point of the trajectory point which needed to calculate the SED. The point *a* and point *b* in Fig. 1 are the datum points of the SED of point *p*.

3 Synchronous Euclidean Distance Limited Thresholds Method

3.1 Thresholds Algorithm

TA sets a motion direction change threshold and a speed change threshold, it predicts the next point's possible area by using the moving object at the current trajectory point's speed and motion direction and two predetermined thresholds. If the next point is in this prediction area means that the amount of this point's information is small, it will be deleted. While it will be retained as the key point if it is outside the prediction area. However, it will result in an evaluating errors problem when a continuous small angle steering occurs. As shown in Fig. 2, all points will be deleted in this case, while it should be retained at least one point because of the cumulative error.

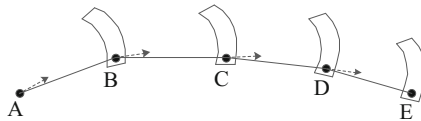


Fig. 2. Continuous small angle steering

In order to avoid this situation, TA uses a combination of two predictions method to infer the next point's possible area. As shown in Fig. 3, the judgment of point D does not only depend on the prediction of B and C, but also depend on the overlapping area of the AB prediction area and the BC prediction area. As the point D is outside this overlapping area, so that the point D is the key point. This method solves the problem of evaluating errors when a continuous small angle steering occurs to some extent. While this problem still exists if the angle is small enough.

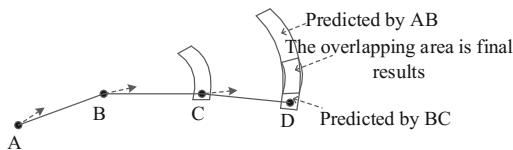


Fig. 3. Threshold algorithm

Another serious drawback of the TA is that the offset is not considered. As shown in Fig. 3, the area predicted by AB is significantly larger than the area predicted by BC because of the time interval of BD is much larger than that of CD. It means the accumulation of time will result in a large offset situation even when a small angle changes occurs. Therefore, it is necessary to evaluate the offset for the GPS trajectory with uncertain time frequency sampling.

3.2 Synchronous Euclidean Distance Limited Thresholds Method

SLTA is an online GPS trajectory data compression algorithm, which evaluates the size of the trajectory point information by the magnitude of the steering angle and the magnitude of the speed change of the trajectory point and uses the SED to limit the offset of the point to achieve the better information retention.

Definition 1. Trajectory vector. The trajectory vector is the vector combined by two different time trajectory points in a trajectory, which starting at a point with a small time and ending with a large time.

Definition 2. Trajectory vector’s time. The trajectory vector’s time is the time difference between the end point and the starting point of the trajectory vector.

Definition 3. Trajectory point’s speed. The trajectory point’s speed is the ratio of modular and time of the trajectory vector starting from this trajectory point. As shown in Fig. 4, the speed of point A on the original trajectory is the ratio of $|AB|$ to t_{AB} , and the speed of point B is the ratio of $|BC|$ to t_{BC} . On the simplified trajectory, the speed of point A is the ratio of $|AD|$ to t_{AD} .

Definition 4. Trajectory point’s motion direction. The trajectory point’s motion direction is the direction of the trajectory vector at which the trajectory point is the starting point. As the Fig. 4 shows, the direction of point A on the original trajectory point is the direction of the vector AB, and the direction of point C is the direction of the vector CD. On the simplified trajectory, the direction of point A is the direction of vector AD.

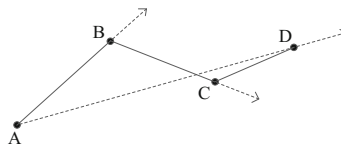


Fig. 4. The speed and motion direction of the trajectory point

SLTA is described as follows:

Algorithm 1 Synchronous Euclidean Distance Limited Thresholds Algorithm (SLTA)

Input: speed change threshold $speed_th$, motion direction change threshold $angle_th$, SED threshold sed_th , original trajectory $P = \{p_1, p_2, \dots, p_n\} = \{(x_1, y_1, t_1), \dots, (x_n, y_n, t_n)\}$

Output: simplified trajectory $P' = \{p_{i1}', p_{i2}', \dots, p_{im}'\} = \{(x_{i1}', y_{i1}', t_{i1}'), \dots, (x_{im}', y_{im}', t_{im}')\}$

```

1) for each point  $p$  in  $P$  do
2)   if  $p$  is the first then
3)      $et\_key\_point(p)$ 
4)      $key\_state = cal\_motion\_state(p)$  //calculate the motion state of point  $p$ 
5)      $last\_key = p$ 
6)     continue
7)   end
8)   if  $p == last\_key + 1$  then
9)      $state = cal\_motion\_state(p)$ 
10)    if  $|state - key\_state| > (speed\_th \text{ or } angle\_th) \text{ or } SED$  (with  $last\_key$  and  $p+1$  as the datum point)  $> sed\_th$  then
11)       $set\_key\_point(p)$ 
12)       $last\_key = p$ 
13)       $key\_state = state$ 
14)      continue
15)    end
16)     $pred\_state = state$ 
17)    end
18)    if  $p \neq last\_key + 1$  then
19)       $state = cal\_motion\_state(p)$ 
20)      if  $|state - key\_state| > (speed\_th \text{ or } angle\_th) \text{ or } |state - pred\_state| > (speed\_th \text{ or } angle\_th) \text{ or } SED$  (with  $last\_key$  and  $p+1$  as the datum point)  $> sed\_th$  then
21)         $set\_key\_point(p)$ 
22)         $last\_key = p$ 
23)         $key\_state = state$ 
24)        continue
25)      end
26)       $pred\_state = state$ 
27)    end
28)  end

```

The algorithm steps are as follows:

(1) Select the first point of the trajectory as the key point and take it as the last key point, then calculate the speed and motion direction of the original trajectory as the reference motion state of the key point. (2) When the first point arriving which is after the key point, calculate its speed and motion direction on the original trajectory, and use the key point reference motion state. If the motion state change of the new point exceeds the prescribed threshold or the new point's SED exceeds the specified threshold, remain it as a new key point, and update the key point reference motion state, otherwise the new point's motion state will be saved as the predecessor reference motion state. (3) When the second and subsequent points arriving after the key point (that is, the first point after the key point which is not evaluated as the key point), calculate its speed and motion direction on the original trajectory, and use the key point reference motion state and the predecessor point reference motion state. If the motion state change of the new point exceeds the specified threshold value or the new point's (consider the last key point and the next point of the new point as the datum point) SED exceeds the specified threshold, remain it as a new key point, and update the key point

reference motion state. SLTA uses the key point reference motion state and the predecessor reference motion state to limit the current motion state, which solves the point's cumulative error and mutation error problem. The SED introduced in this kind of algorithm make sure that the problem of offset large keys will not being ignored caused by micro-variable state of motion and too long movement time. SLTA ensure the retention of multiple trajectory point's information by limiting the three thresholds of speed change threshold, motion direction change threshold and SED threshold.

4 Experiments Study and Analysis

In this section, we develop a GPS trajectory data compression prototype system to evaluate the compression effect and performance of SLTA. It is worth mentioning that all of the experiments were run on a commodity computer with Intel Core i5 CPU (2.3 GHz) and 4 GB RAM. We use a real-world large scale trajectory dataset from the Geolife [12] project in our experiment.

4.1 The Comparative Algorithms

As an improved algorithm of TA, SLTA and TA are both belong to the online GPS trajectory data compression algorithm based on the change of motion state. SLTA not only can set the speed threshold and angle threshold, but also can set the SED threshold, one more parameter than TA, so TA is used as the main contrast algorithm of SLTA. In addition, SLTA is compared with other algorithm based on offset calculation in this paper: TD-TR, OPW-TR and SQUISH-E algorithm. Among them, SQUISH-E algorithm can set the two parameters of compression ratio and SED, so the SQUISH-E algorithm is divided into two alignment methods in this paper, the first is to set its compression rate to 1, expressed as SQUISH-E (*sed*), the second is to set the threshold of its SED to 0, expressed as SQUISH-E (*ratio*). When the compression rate is set to 1, the type of SQUISH-E (*sed*) is an offline algorithm.

4.2 Experimental Analysis

The definitions used in the experiment are as follows:

Definition 5. Compression ratio. The compression ratio is the ratio of the number of trajectory points after compression to before compression.

Definition 6. The average SED error. The average SED error can be calculated as follows formula:

$$\overline{SED} = \left(\sum_{i=1}^n SED(p_i) \right) / n \quad (4)$$

p_i denotes the i -th point on the original trajectory, n represents the number of the points of the original trajectory, $SED(p_i)$ represents the SED of p_i in terms of the adjacent predecessor and successor point of p_i on the simplified trajectory.

Definition 7. The average angle error. The average angle error is the average value of the angles from the motion direction of each point in the original trajectory and the trajectory vector of predecessor point and the successor point on the simplified trajectory corresponding to this point. The smaller the average angle error, the better the compression effect of the algorithm.

Definition 8. The average speed error. The average speed error is the average value of the speed of each point in the original trajectory and the absolute value of the speed difference of the predecessor point in the simplified trajectory corresponding to this point. The smaller the average speed error, the better the compression effect of the algorithm.

In this paper, we choose 1 m, 2 m, 3 m, 4 m, 5 m, 6 m, 10 m, 20 m and 40 m as parameters when we use the SED threshold as the algorithm parameter. The applicable algorithms are TD-TR, OPW-TR, and SQUISH-E (*sed*). When the compression ratio is used as the algorithm parameter, we select 2, 3, 4 and 10 as parameters. The applicable algorithms is SQUISH-E (*ratio*). Table 1 shows the results of the comparison test with the TA when the SED threshold of the SLTA is set to infinity.

Table 1. The comparison result of SLTA and TA

Method	Speed threshold (km·h ⁻¹)	Angle threshold (°)	Compression ratio (%)	The average SED error (m)	The average speed error (m·s ⁻¹)	The average angle error (°)
SLTA	5	3	66.15	0.453447	10.7399	1.829455209
SLTA	5	5	60.70	0.672841	10.7387	2.036477183
SLTA	5	7	57.05	0.897384	10.7373	2.248501238
SLTA	10	3	61.73	1.10891	10.7381	1.989561904
SLTA	10	5	55.04	1.7451	10.7354	2.249372162
SLTA	10	7	50.56	2.44565	10.7326	2.51991426
SLTA	20	3	59.72	2.4585	10.7324	2.045931288
SLTA	20	5	52.27	4.14454	10.7254	2.338429828
SLTA	20	7	47.25	5.75292	10.7185	2.648192136
TA	5	3	77.93	125.593	10.8688	0.703620311
TA	5	5	72.82	133.699	10.8464	1.210251458
TA	5	7	69.54	138.253	10.8292	1.656754499
TA	10	3	73.77	219.265	10.8603	0.984252517
TA	10	5	66.79	289.571	10.7877	1.957962276
TA	10	7	62.21	359.777	10.7078	2.858543101
TA	20	3	72.25	272.479	10.8267	1.098882105
TA	20	5	64.42	403.911	10.7021	2.28146455
TA	20	7	59.17	968.093	10.5445	3.443671967

As can be seen from Table 1, the average speed error of SLTA and TA is very small and almost negligible; SLTA compression ratio is generally lower than the TA

about 10%, which indicates that compression effect of SLTA is better than TA; The average SED error of SLTA is much less than TA. This is because the TA does not solve the problem of cumulative error. When the number of trajectory points increasing, in the moving trajectory where exits the cumulative error, the cumulative error will be increasing with the increase of the number of points. While SLTA uses a key point reference motion state, which can make an evaluation to the cumulative error after the key point. When the cumulative error reaches the threshold, the new key points are retained, so as to avoid the expansion of the cumulative error. Therefore, the effect of this cumulative error will not occur which greatly enhancing the stability of SLTA. When the speed threshold is 10 km/h and the angle threshold is more than than 6° , the average angle error of SLTA is less than TA; The average angle error of SLTA is significantly smaller than that of TA with the increase of angle threshold, which also shows that the stability of SLTA is higher than that of TA, because with the increase of angle threshold, the situation of small angle continuous turning of TA will increase more and more and correspondingly, the growth of average angle error of TA is faster than SLTA.

It can be seen from Fig. 5 that the average angle error of SLTA is the smallest ($1.5^\circ\text{--}2.3^\circ$), and the retention of motion direction information is better than the compression algorithm of GPS trajectory data based on the offset calculation, which reflects the effectiveness of the SLTA algorithm.

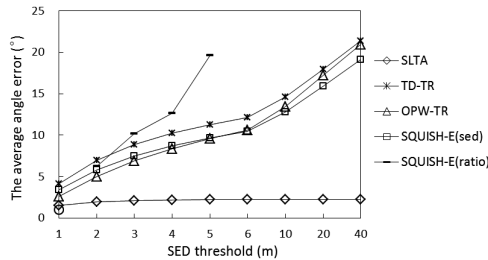


Fig. 5. The comparison of average angle error of SLTA and other algorithms

5 Conclusions

In this paper, we propose a SLTA to do trajectory data compression. SLTA improves the TA by modifying its reference motion state, adding the key point reference motion state and the key-predecessor reference motion state, and limiting the SED. It can solve the problem of insufficient consideration to the cumulative error and offset which online GPS trajectory data compression based on motion state change and the key point insufficient evaluation of online GPS trajectory data compression based on the offset calculation. SLTA could reach better information retention. The experiment results show that the SLTA algorithm retains more trajectory information while performing trajectory compression effectively. In different traffic modes, the performance of the same GPS trajectory data compression algorithm is different, so the combination of traffic pattern recognition and GPS trajectory data compression will be the focus of the next research.

References

1. Chen, M., Xu, M., Franti, P.: Compression of GPS trajectories. In: Proceedings of the 2012 Data Compression Conference (DCC), pp. 62–71. IEEE, Piscataway (2012)
2. Chen, M., Xu, M., Franti, P.: Compression of GPS trajectories using optimized approximation. In: Proceedings of the 2012 21st International Conference on Pattern Recognition (ICPR), pp. 3180–3183. IEEE, Piscataway (2012)
3. Douglas, D.H., Peucker, T.K.: Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: Int. J. Geograph. Inf. Geovis.* **10** (2), 112–122 (1973)
4. Hershberger, J., Snoeyink, J.: Speeding up the Douglas-Peucker Line-Simplification Algorithm, pp. 134–143. University of British Columbia, Department of Computer Science (1992)
5. Hershberger, J., Snoeyink, J.: An $O(n \log n)$ implementation of the Douglas-Peucker algorithm for line simplification. In: Proceedings of the Tenth Annual Symposium on Computational Geometry, pp. 383–384. ACM, New York (1994)
6. Agarwal, P.K., Har-Peled, S., Mustafa, N.H., et al.: Near-linear time approximation algorithms for curve simplification. *Algorithmica* **42**(3–4), 203–219 (2005)
7. Ma, J., Xu, S., Pu, Y., et al.: A real-time parallel implementation of Douglas-Peucker polyline simplification algorithm on shared memory multi-core processor computers. In: Proceedings of the 2010 International Conference on Computer Application and System Modeling (ICCAASM), pp. V4-647–V4-652. IEEE, Piscataway (2010)
8. Keogh, E., Chu, S., Hart, D., et al.: An online algorithm for segmenting time series. In: *ICDM 2001: Proceedings IEEE International Conference on Data Mining*, pp. 289–296. IEEE, Piscataway (2001)
9. Muckell, J., Hwang, J.H., Patil, V., et al.: SQUISH: an online approach for GPS trajectory compression. In: Proceedings of the 2nd International Conference on Computing for Geospatial Research and Applications, p. 13. ACM, New York (2011)
10. Muckell, J., Olsen Jr., P.W., Hwang, J.H., et al.: Compression of trajectory data: a comprehensive evaluation and new approach. *GeoInformatica* **18**(3), 435–460 (2014)
11. Meratnia, N., de By, R.A.: Spatiotemporal compression techniques for moving point objects. In: Bertino, E., et al. (eds.) *EDBT 2004*. LNCS, vol. 2992, pp. 765–782. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-24741-8_44
12. Microsoft Research. Geolife GPS Trajectories data sample. [EB/OL], 09 Aug 2012. <http://research.microsoft.com/en-us/downloads/b16d359d-d164-469e-9fd4-daa38f2b2e13/>



Mining Temporal Discriminant Frames via Joint Matrix Factorization: A Case Study of Illegal Immigration in the U.S. News Media

Qingchun Bai¹, Kai Wei², Mengwei Chen¹, Qinmin Hu³(✉), and Liang He¹

¹ School of Computer Science and Software Engineering,
East China Normal University, Shanghai, China

qchbai@ica.stc.sh.cn, cmwwmcrriyy@outlook.com, lhe@cs.ecnu.edu.cn

² School of Social Work, University of Pittsburgh, Pittsburgh, USA
Kai.Wei@pitt.edu

³ Department of Computer Science, Ryerson University, Toronto, Canada
vivian@ryerson.ca

Abstract. Framing detection has emerged to be an important topic in recent natural language processing research. Although several frameworks have been proposed, little is known about how to detect temporal discriminant frames. This study proposes a framework for discovering temporal discriminant frames, with a focus on identifying emergent frames in news discussions of illegal immigration issue. Built on joint non-negative matrix factorization (NMF), we propose the njNMF algorithm, an improved joint matrix factorization algorithm, to detect the **temporal** frames. We conducted experiments using the njNMF algorithm to identify emergent frames. The results of our experiments show that framing of illegal immigration changes over time, from human trafficking frames, to more recent economic and criminality frames. These findings suggest the utility of our temporal framing approach and can be used as a framing detection tool for policy researchers to understand the role of news framing in public agenda setting.

Keywords: Joint NMF · Temporal discriminant frame
Framing evolution

1 Introduction

One way that news media influences the public is through framing - portraying a policy issue from one perspective to exclude alternative perspectives. Given an immigration issue, what frames are used by news media can set public agenda by covering certain perspectives of the issue frequently and prominently [9]. In this way, the audience will regard these perspectives as more important compared to those that are not frequently reported in news media. Among all immigration issues in the United States, illegal immigration is the most controversial

issue. The way news media portrays illegal immigration issue can have significant influence on public attitudes toward the issue and related policy agenda. For example, the U.S. news media can frame illegal immigration in the following ways:

1. Impact on economy: “Some economists say that immigrants, legal and illegal, produce a net economic gain, while others say that they create a net loss”.
2. Impact on family: “More and more immigrant families who come to the southern border seeking asylum are being charged in federal criminal courts from El Paso to Arizona”.

Consider these news frames above, different news framing of an issue will have differential impact on how the public perceive immigrants as well as their support for immigration policy agenda. Recent NLP researchers have started to focus on automatic detection of frames [3, 10]. However, this line of research is carried out in a single dimension and cannot automatically detect distinct frames and similar frames over time from multiple news sources.

In this paper, we aim to take analysis on the issue of illegal immigration from temporally sorted news articles to: **discover temporal discriminant frames**. To achieve these goals, we propose a novel framework that can automatically detect temporal framing of an policy issue, e.g. illegal immigration, over time. First, we design an emergent framing algorithm that builds on an improved joint NMF algorithm. To detect most similar and most distinct frames of a given issue across multiple data sources, penalty functions are incorporated into the joint NMF to control distinct and similar frames. Second, we carry out extensive experiments to demonstrate the efficiency of our approach and illuminate its utility by exploring the temporal framing of illegal immigration in the U.S. news media.

2 Related Work

Framing is a way to choose “a few elements of perceived reality and assembling a narrative that highlights connections among them to promote a particular interpretation” [5]. Past NLP research has focused on framing related tasks, including automatic detection on sentiment, stance, and topics [6]. While these works are related to framing detection, these methods do not address framing scholar’s concerns “with persistent patterns of representation of particular issues— without necessarily tying these to the states or intentions of authors—and the effects that such patterns may have on public opinion and policy” [3].

Given the needs of framing detection, recent NLP scholars started to focus on automatic detection of frames in news and social media. Card et al. composed news media frames corpus, which contains 3 issues (Immigration, Smoking and Same-sex marriage) and about 4,000 articles [2]. Following this study, Card et al. proposed an unsupervised model for identifying frames in a collection of news articles about immigration by using a Dirichlet persona based model, and conducted experiments that showcased its utility [3]. More recently, Naderi et al.

employed LSTM classification methods to classify in news articles, and the model achieved good performance [10]. Johnson et al. defined the political frames and the ideological phrase indicators to analyze the general frames of politicians [7]. Dehghan et al. focused on the problems of detecting news bias in different media sources [4].

These prior works on framing detection have focused on independent issues, such as classifying pre-defined news frames and detecting community frames, few of them also have focused on distinguishing different media reports. Therefore, we propose a novel framework that can automatically detect framing of an issue over time.

3 Our Approach to Framing Detection

3.1 Problem Statement

Given a document set D sorted by time steps $t = (1, 2, \dots, Z)$, our goal is to discover distinct and similar frames related to an issue. Here $D = \sum_{t=1}^Z D_t$ and $Z \geq 2$. To tackle the temporal distinct frame, our strategy is to detect all latent frames from subsets D_t to D_{t+1} , we then discover distinct and similar frames from the two subsets. One of the key issues is to determine the proportion of similar frame V_c as well as distinct frame V_d in V_k . Here, *feature matrix* of subset D_t is represented as *term-document matrix* X_t . *Frame matrix* V_t extracted from subset D_t is defined as $V_t = \{S_1, S_2, \dots, S_h, \dots, S_{n_t}\}$, here S_h is a *frame content vector*. Parameter n_t is the number of frames contained in the subset D_t .

3.2 An Improved Joint Non-negative Matrix Factorization

Our proposed framework incorporates an improved joint NMF algorithm (namely, *njNMF*) by adding a jointed penalty function into the standard NMF algorithm. Using the njNMF algorithm, we can extract temporally distinct and similar frames from two datasets. The njNMF algorithm simultaneously decomposes two feature matrices, X_{t1} and X_{t2} , using

$$\begin{cases} X_{t1} \approx W_{t1} \times H_{t1}^\top \\ X_{t2} \approx W_{t2} \times H_{t2}^\top \end{cases} \quad (1)$$

After the decomposition step, we can get two base matrices W_{t1} and W_{t2} , and two coefficient matrices H_{t1} and H_{t2} , respectively. We set base matrix $W_{i,c}$ as similar frames, and matrices $W_{i,d}$ as discriminate frames. We illustrate the decomposing procedure in the right-up of Fig. 1.

Penalty Functions. To control the decomposition of base matrices W_i , we introduce two penalty functions $\mathcal{R}_1(W_{1c}, W_{2c})$ and $\mathcal{R}_2(W_{1d}, W_{2d})$ into the joint matrix decomposing procedure. We define the objective function as

$$\min_{\substack{W_1, H_1, \\ W_2, H_2 \geq 0}} \lambda_1 \mathcal{F}_1(W_1, H_1^\top) + \lambda_2 \mathcal{F}_2(W_2, H_2^\top) + \omega_1 \mathcal{R}_1(W_{1,c}, W_{2,c}) + \omega_2 \mathcal{R}_2(W_{1,d}, W_{2,d}). \quad (2)$$

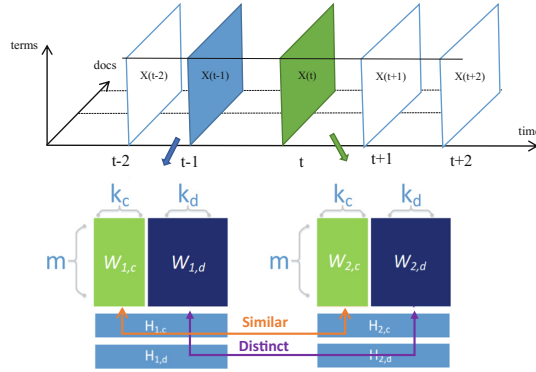


Fig. 1. The framework of temporally discriminant frames detection.

Here, function \mathcal{F}_1 and \mathcal{F}_2 are both defined as $\|\bullet\|_F^2$. Through Eq. (2), we can find the proper value of k to determine the final extracted frames, there is $k = k_c + k_d$, where k_c is the number of the similar frames between D_i and D_j , and k_d stands for the number of distinct frames. The smaller \mathcal{R}_1 and \mathcal{R}_2 stand for the better similar (distinct) frames of topics. Here, \mathcal{R}_1 and \mathcal{R}_2 denote as

$$\begin{cases} \mathcal{R}_1(W_{1,c}, W_{2,c}) = \mathcal{N}\mathcal{F}_t(W_{1,c} - W_{2,c}) & (3) \\ \mathcal{R}_2(W_{1,d}, W_{2,d}) = \mathcal{S}\mathcal{F}_t(W_{1,d} \times W_{2,d}) & (4) \end{cases}$$

where subscript t indicates that the matrices $W_i = [W_{i,c} \ W_{i,d}]$ and $H_i = [H_{i,c} \ H_{i,d}]$ belong to different time points.

Optimization Algorithm. In order to get an optimal k and to find similar and distinct frames between datasets D_1 and D_2 , we need to get an optimal solution of Eq. 2. Our optimization algorithm has three parts: (1) Minimize penalty function $\mathcal{R}_1(W_{1,c}, W_{2,c})$ to compute two matrices $W_{1,c}$ and $W_{2,c}$, which represent the similar frames of two datasets D_1 and D_2 . (2) Minimize penalty function $\mathcal{R}_2(W_{1,d}, W_{2,d})$ to compute two matrices $W_{1,d}$ and $W_{2,d}$. The two matrices refer to distinct frames of datasets D_1 and D_2 . (3) Select proper values for parameters ω_1 and ω_2 to balance the weight between $\mathcal{R}_1(W_{1,c}, W_{2,c})$ and $\mathcal{R}_2(W_{1,d}, W_{2,d})$.

Originated in a block-coordinate descent framework [8], we use an coordinate descent algorithm of iterative optimization of Eq. (2) using

$$\begin{aligned} w_{1,c}^l &\leftarrow \frac{H_1^\top H_1}{H_1^\top H_1 + n_1} w_{1,c}^l + \alpha(w_{2,c}^l \times h_2^l) \\ w_{2,c}^l &\leftarrow \frac{H_2^\top H_2}{H_2^\top H_2 + n_2} w_{2,c}^l + (1 - \alpha)(w_{1,c}^l \times h_1^l) \end{aligned} \tag{5}$$

where $(\cdot)^l$ represents the l -th column of a matrix in parentheses, the parameter α controls the weight of result from the correcting function during the iteration. The updating rules for $w_{2,c}^l, w_{2,d}^l$ can also be derived in a similar manner.

4 Experiment

4.1 Data Collection

Data were collected from LexisNexis using the search term “illegal w/1 immigrant” (illegal with one space to immigrant). LexisNexis is one of the largest newspaper archive in the world which contains major US newspapers (e.g. the New York Times and Washington Post). In total, we extracted 13,039 newspaper articles from 1997 to 2011 ($n = 13,039$). Figure 2 shows the distribution of illegal immigration related news articles from 1997 to 2011. We use the annual number of the news reports with average 900 documents to ensure the topic balance in the news data. We pre-processed the data using the following steps: removing stop words, stemming, removing meaningless words, and others. Following the data pre-processing, we used the TF-IDF, which can compute how important a word is to a document, where $tf_{ij} = \frac{n_{i,j}}{\sum n_{k,j}}$, $idf_i = \log \frac{|D|}{|j:t_i \in d_j|}$, and $value = tf_{ij} \times idf_i$.

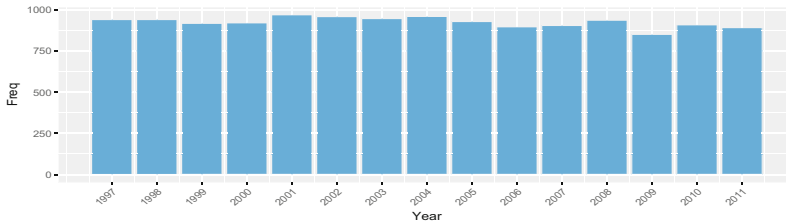


Fig. 2. Number of news articles containing “illegal w/1 immigrant” from 1997 to 2011.

4.2 Results and Analysis

The njNMF algorithm was used to analyze temporal framing of illegal immigration in the U.S. news media from 1997 to 2011. Similar and distinct frames and their relevant terms were extracted from the data.

Temporal Framing of Illegal Immigration. Figure 3 shows four distinct frames in the issue of illegal immigration in the news media over time: legitimacy regarding citizenship (Fig. 3(a)), human trafficking (Fig. 3(b)), travel behavior (Fig. 3(c)), and economic concerns (Fig. 3(d)).

Figure 3(a) shows that news media framing of illegal immigration focused on legitimacy of citizenship, indicated by words such as “legally”, “illegally”, “deadline”, and “deport”. The discussion of the “legitimacy” topic reaches a peak about 2004, the topic decreases as the news media shifted their focus in illegal immigration in recent years. This pattern follows a consistent trend as the illegality and criminality framing of Latino immigrants found in previous study [11]. Figure 3(b) reveals the framing of illegal immigration from then human trafficking perspective. For example, we noticed that there were more words such as

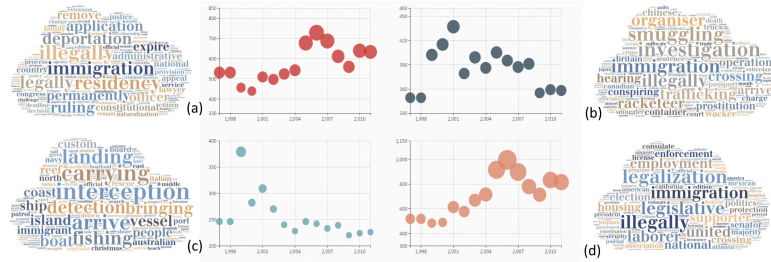


Fig. 3. Examples of similar frames evolution. Four different colors represent four frames in the trend charts. The y-axis refers to the distribution of the topic in each year, which is the x-axis. (Color figure online)

“smuggle”, “truck”, “passport”, “death”, and “cargo” appearing in news coverage of illegal immigration from 1997 to 2004. The temporal pattern of this frame seems to be aligned with the overall human trafficking trend during this period, which peaked in 2001, and then decreased from 2001 to 2004. Additionally, it seems that during the early years (from 1997 to 1998), the framing of illegal immigration focuses on human traffic aspects of illegal immigration, indicating by the word “smuggling”. Figure 3(c) presents the emergent framing of immigration that focuses on immigrant’s travel behavior, indicated by words such as “boat”, “vessel”, and “landing”. However, as the news media attention focused more on economic aspects of illegal immigration in recent years, the focus of this frame in news coverage of illegal immigration declined.

Figure 3(d) shows economic concerns of illegal immigration, suggested by words such as “license”, “industry”, “laborer”, “housing”, and “employment”. This frame reflects mixed attitudes about immigration in news media, i.e., a concern about whether immigration, especially illegal immigration, makes economic contribution or poses economic threat. On one hand, the arrival of immigrant population could address the issue of labor shortage and decrease the cost of industry. On the other hand, the influx of immigrants, especially undocumented immigrants, could potentially increase the economic burden such as housing. The economic aspects of immigration in the U.S. historically has been the major focus, and it is still a primary interest regarding immigration issue. This may explain why this framing of illegal immigration issue occupies the highest attention among all the revealed frames. The peak is aligned with the framing of Latino immigrants as threat discovered in previous work [11].

Distinct Frames in U.S. News Media over Time. We also analyzed the discovered distinct frames and their relevant terms in Fig. 4. We found three distinct frames of illegal immigration: (1) criminalization (red letters), (2) federal government regulation (blue letters), and (3) economic concerns (green letters).

Figure 4(a) shows that news media has increasingly focused on the framing of immigrants as criminals, i.e., criminalization of immigration. This frame is indicated by the terms that show more negative attitudes such as “illegal”, “crime”,

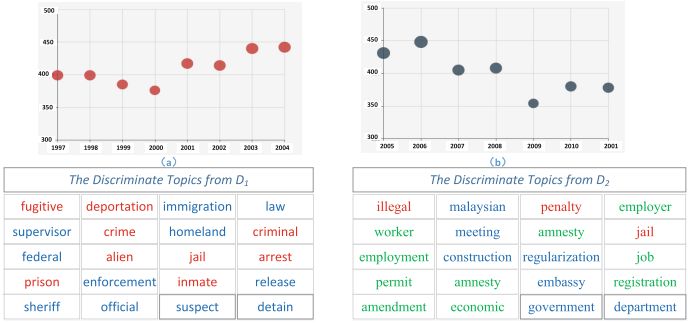


Fig. 4. Distinct frames in U.S. news media over time. Terms in green show economic frame, terms in red indicate criminalization frame, and terms in blue mean federal government regulation frame. (Color figure online)

“jail”, and criminal (see red letters in Fig. 4). The criminalization of immigration has been long argued in social science research (e.g., Ackerman et al. [1]). One of the major consequences of criminalization of immigration is the increase of prison industry and deportation. Our study observed this frame and found terms that indicate this frame such as “deportation”, “jail”, “penalty” and “prison” (see red letters in Fig. 4).

Figure 4(b) shows the economic concerns about immigration. This aspect is indicated by the topic terms such as “employer”, “economic” and “job” (green letters). The economic concern about immigration seems to be a major frame in the news media discussion of immigration because it occurred in both our findings (see discussion about temporal framing). This result indicates that whether immigration makes economic contribution or poses economic threat can be a central concern in setting public agenda.

5 Conclusion and Future Work

In this study, we propose a framework to discover temporal discriminant frames from temporally sorted document corpus. We conducted experiments in a collection of news articles discussing the issue of illegal immigration and identified four distinct emergent frames: legitimacy regarding citizenship, human trafficking, travel behavior, and economic impact frames. We found that as the news media attention focused more on economic aspects of illegal immigration in recent years, the focus of this frame in news coverage of illegal immigration declined. This result indicates that framing of illegal immigration changes over time. This change might reflect the different focus of media in setting public agenda in illegal immigration issue over time. Future research might consider a more fine-grained approach to framing detection by comparing different temporal framing of illegal immigration by different news outlets. In summary, the findings of our study suggest the utility of our temporal framing approach and

can be used as a automatic framing detection tool for social science researchers to explore how news media set public agenda.

Acknowledgments. This research is funded by the Natural Science Foundation of Shanghai (No. 17ZR1444900), National Natural Science Foundation of China (No. 41601418), Scientific technological research project of Henan Province (No. 172102210539).

References

1. Ackerman, A.R., Furman, R.: The criminalization of immigration and the privatization of the immigration detention: implications for justice. *Contemp. Justice Rev.* **16**(2), 251–263 (2013)
2. Card, D., Boydston, A.E., Gross, J.H., Resnik, P., Smith, N.A.: The media frames corpus: annotations of frames across issues. In: *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, vol. 2, pp. 438–444 (2015)
3. Card, D., Gross, J., Boydston, A., Smith, N.A.: Analyzing framing through the casts of characters in the news. In: *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 1410–1420 (2016)
4. Dehghan, A., Montgomery, L., Arciniegas-Mendez, M., Ferman-Guerra, M.: Predicting news bias
5. Entman, R.M.: Framing bias: media in the distribution of power. *J. Commun.* **57**(1), 163–173 (2007)
6. Feldman, R.: Techniques and applications for sentiment analysis. *Commun. ACM* **56**(4), 82–89 (2013)
7. Johnson, K., Lee, I.-T., Goldwasser, D.: Ideological phrase indicators for classification of political discourse framing on Twitter. In: *Proceedings of the Second Workshop on NLP and Computational Social Science*, pp. 90–99 (2017)
8. Kim, H., Choo, J., Kim, J., Reddy, C.K., Park, H.: Simultaneous discovery of common and discriminative topics via joint nonnegative matrix factorization. In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 567–576. ACM (2015)
9. McCombs, M.E., Shaw, D.L.: The agenda-setting function of mass media. *Public Opin. Q.* **36**(2), 176–187 (1972)
10. Naderi, N., Hirst, G.: Classifying frames at the sentence level in news articles. *Policy* **9**, 4–233 (2017)
11. Wei, K., Lin, Y.-R.: The evolution of Latino threat narrative from 1997 to 2014. In: *ICConference 2016 Proceedings* (2016)



Enhancing Cluster Center Identification in Density Peak Clustering

Jian Hou¹(✉), Aihua Zhang¹, Chengcong Lv¹, and Xu E^{2,3}

¹ College of Engineering, Bohai University, Jinzhou 121013, China
dr.houjian@gmail.com

² College of Information Science, Bohai University, Jinzhou 121013, China

³ College of Food Science and Technology, Bohai University, Jinzhou 121013, China

Abstract. As a clustering approach with significant potential, the density peak (DP) clustering algorithm is shown to be adapted to different types of datasets. This algorithm is developed on the basis of a few simple assumptions. While being simple, this algorithm performs well in many experiments. However, we find that local density is not very informative in identifying cluster centers and may be one reason for the influence of density parameter on clustering results. For the purpose of solving this problem and improving the DP algorithm, we study the cluster center identification process of the DP algorithm and find that what distinguishes cluster centers from non-density-peak data is not the great local density, but the role of density peaks. We then propose to describe the role of density peaks based on the local density of subordinates and present a better alternative to the local density criterion. Experiments show that the new criterion is helpful in isolating cluster centers from the other data. By combining this criterion with a new average distance based density kernel, our algorithm performs better than some other commonly used algorithms in experiments on various datasets.

Keywords: Clustering · Density peak · Local density · Cluster center

1 Introduction

Data clustering has wide applications in such fields as data mining, pattern recognition and others. Many clustering algorithms of different types have been developed and some of them have generated impressive results in application. Some commonly used algorithms include k-means, spectral clustering [13, 16], DBSCAN [7], mean shift [5] and their variants. Recently, some new algorithms have been proposed, including affinity propagation (AP) [3], robust spectral clustering [19], dominant sets (DSets) [14]. Noticing that many algorithms require to determine the number of clusters beforehand, [8, 12] have presented some methods to solve this problem. Since some algorithms detect only spherical clusters, density based algorithms have received a lot of attention [1, 2].

In density based clustering algorithms, DBSCAN relies on a density threshold to detect cluster borders, and the density threshold is represented by two parameters $MinPts$ and Eps . While DBSCAN has been shown to perform well in many experiments, it may not be easy to determine the appropriate parameters. In addition, a fixed set of parameters imply a fixed density threshold, which may not be appropriate for datasets where cluster densities vary significantly. Different from DBSCAN-like algorithms, the density peak (DP) algorithm presented in [15] accomplishes the clustering process on the basis of density relationship. By treating local density peaks as the candidates of cluster centers, the DP algorithm finds that cluster centers have both great ρ 's and great δ 's, and either the ρ 's or δ 's of non-density-peak data are small. This algorithm then uses both ρ and δ , or $\gamma = \rho\delta$, to identify cluster centers, and then group the other data into clusters based on density relationship among neighboring data. Different from cluster centers surrounded by data of smaller density, non-density-peak data usually have the nearest neighbors greater density in the neighborhood, corresponding to small δ 's. Consequently, the distance δ is effective in isolating cluster centers from the non-density-peak data. While cluster centers usually have greater local density than the neighboring non-density-peak data, the density of non-density-peak data may not be small in absolute magnitude. In other words, non-density-peak data may have great or small local density, and the ρ criterion is not very informative in strengthening the specificity of cluster centers. For the purpose of solving this problem, we study the cluster center identification process and find that the role of density peak is more important for a cluster center than a great density. We then present an enhanced criterion based on local density of subordinates to replace the original local density ρ . Furthermore, a new density kernel is proposed to overcome the drawbacks of the cutoff and Gaussian kernels. By combining the new criterion and density kernel, our algorithm performs well in experiments and compares favorably to some commonly used and recently proposed algorithms.

2 Density Peak Clustering Algorithm

An attractive property of density based clustering algorithms is that they detect non-spherical clusters. While the DBSCAN algorithm based on a density threshold, the DP algorithm makes use of the density information in a different manner. We use examples to demonstrate how the DP algorithm identify cluster centers and accomplish the clustering process. With the Aggregation [10] dataset, we calculate ρ in the first step. The cutoff kernel defines the local density as the count of data in the d_c -radius neighborhood, where d_c is the cutoff distance to be specified. The distribution of ρ and δ of all the data is shown in Fig. 1(a). Obviously only a few data have both great ρ and great δ and the majority of the data have either small δ 's or small ρ 's. This makes it feasible to determine cluster centers by selecting the data of great ρ 's and great δ 's. Noticing that with Fig. 1(a) two thresholds are necessary to determine cluster centers, we sort the data according to $\gamma = \rho\delta$ in the decreasing order and show the distribution of γ

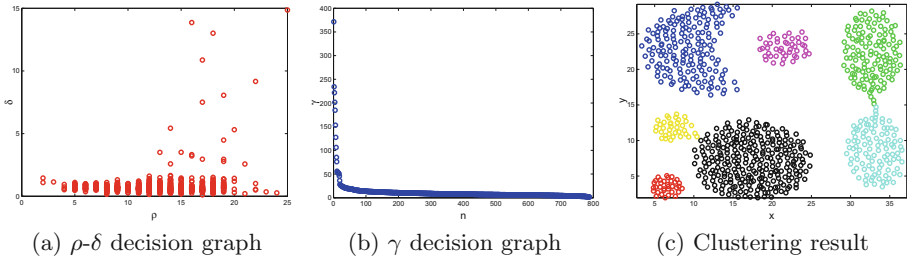


Fig. 1. Decision graphs and clustering results with the cutoff kernel.

in Fig. 1(b), where a few data are with significantly greater γ than the others. With the γ decision graph we only need one threshold to select out the cluster centers.

With the cluster centers available, the DP algorithm determines the labels of the non-density-peak data based on data density relationship. The clustering of non-density-peak data is on the basis of the assumption that one data and the nearest neighbor of greater local density are in the same cluster. With this assumption, the non-density-peak data can be assigned labels in the decreasing order of their local density. This process involves only one scan of the data and can be accomplished efficiently. While no proof shows that this assumption holds in theory, the method works well in practice, as shown in the clustering results in Fig. 1(c).

While Fig. 1 shows that cluster centers usually have great ρ , great δ and great γ , they also indicate that it may not be easy to select out the cluster centers with only the decision graphs. The reason is that the differences between great and small ρ 's, δ 's and γ 's are not significant in many cases. For the purpose of avoiding the influence from inappropriate thresholds, we specify the number of clusters in this paper, and the data of the greatest γ 's are identified as cluster centers.

3 Our Approach

Since cluster centers typically have both great ρ 's and great δ 's, we often uses $\gamma = \rho\delta$ to identify cluster centers. As density peaks, cluster centers are surrounded by neighboring data of smaller local density. As a result, they are distant from the nearest data of greater local density and have great δ 's. However, the nearest neighbors of cluster centers may have only slightly smaller density than the correspondingly cluster centers. In other words, non-density-peak data may also have great density. As Fig. 1(a) shows, the local densities of the data are distributed quite evenly, and there are a large amount of data with great local density. This observation indicates that the local density ρ is not as informative as δ in isolating cluster centers from non-density-peak data. In our opinion, this also explains why the DP clustering results are influenced by density kernel types and kernel parameters significantly.

In order to relieve the problems resulted by the uninformative ρ criterion in identifying cluster centers, we make a further study of the cluster center identification process. One intention of the DP algorithm is to use some measures to strengthen the specificity of cluster centers. Since cluster centers have both great ρ 's and great δ 's, and either the ρ 's or δ 's of non-density-peak data are small, the product $\gamma = \rho\delta$ is used as the cluster center identification criterion. We have observed that δ is indeed effective for cluster center identification, and ρ is not so informative in comparison. However, if we remove ρ and uses only δ to select cluster centers, it is likely that the outliers of datasets which are far from other data are identified as cluster centers. In other words, the local density ρ is still effective in preventing outliers from being identified as cluster centers. Therefore instead of removing ρ completely, we propose to enhance the discriminative ability of ρ .

In the DP algorithm, one important feature of cluster centers is that they are surrounded by non-density-peak data of smaller local density. Here we see that what differentiates cluster centers from non-density-peak data is not the great density in absolute magnitude, but the role of density peaks. That is to say, it doesn't matter if one cluster center has a great density, but it matters if it has a greater local density than the neighboring data. Hence we propose to use a criterion measuring the role of density peaks to replace ρ in identifying cluster centers.

In the following we take one data i for example, and denote the cluster containing i as C_i . If i is the cluster center of C_i , it should be surrounded by neighboring data of smaller density. Intuitively we can use the number N_n of neighboring data with smaller density to measure the role of i being the cluster center. A larger N_n means a larger possibility of i being the cluster center of C_i . However, it is possible that the neighboring data with smaller density contain not only the data in C_i , but also some data in other clusters. In this case, N_n cannot measure the possibility of i being the cluster center accurately, and we need to consider only the data in C_i . However, before the clustering is accomplished, the cluster membership of C_i is unknown.

We present the following method to make use of only the data in C_i before the cluster membership is available. It is assumed in the DP algorithm that one data and the nearest data of greater local density are in the same cluster. If one data n is the nearest neighbor of greater local density of data m , we call n as the *superior* of m , and m as the *subordinate* of n , and denote this relationship by $m \rightarrow n$. Evidently one data and all its subordinates should be in the same cluster. Since cluster centers are density peaks, they usually have a large amount of subordinates. On the contrary, the number of subordinates of non-density-peak data may be small or zero. Therefore for the data i , we can use the number N_s of subordinates to measure the probability of i being the cluster center. Furthermore, the local density of subordinates also plays a role in measuring the possibility. In summary, we use the sum of local density of the subordinates to measure the possibility of i being the cluster center, and define the enhanced version of ρ as

$$\eta_i = \sum_{j \in S, j \neq i} \zeta(i, j) \rho_j, \quad (1)$$

where

$$\zeta(x, y) = \begin{cases} 1, & y \rightarrow x, \\ 0, & \textit{otherwise}. \end{cases} \quad (2)$$

Then we can use η to replace ρ and identify cluster centers based on $\gamma' = \eta\delta$. It is worth mentioning that we only use η in determining the cluster centers. The original ρ is still used in grouping non-density-peak data on the basis of density relationship, as it measures the density relationship among neighboring data more accurately.

In addition, we make use of the average distance to a limited amount of nearest data to evaluate the local density. The density kernel obtained this way is presented as a compromise between the cutoff and Gaussian kernels. The cutoff kernel makes use of only the count of data in a neighborhood and discards the distance information to these data. This information loss may influence the local density precision. While the Gaussian kernel makes use of the distance information, it takes into account both the nearest neighbors and the farthest data. In this case, the density kernel may measure the distribution of data in a large region but not a small neighborhood, if the parameter d_c is not selected appropriately. Between these two extremes, our new kernel makes use of the distance to a limited number of neighboring data, and is shown to perform well in experiments.

4 Experiments

In our work η is presented as an enhanced version of local density ρ to improve the discriminative ability, and then use a new density kernel to overcome the drawbacks of existing ones. In this part we firstly validate the effectiveness of the enhanced local density criterion. After that, the whole algorithm is tested and compared with existing commonly used and recently proposed algorithms.

4.1 Enhanced Local Density

The ρ - δ decision graph in Fig. 1 shows that the distribution of data in the range of the local density ρ is quite even, indicating that ρ is not very informative in strengthening the specificity of cluster centers. We are motivated to replace ρ by η to help isolate cluster centers from non-density-peak data. Here we test if η really works in serving this purpose. By replacing ρ with η , we show the η - δ decision graphs and the corresponding ρ - δ decision graphs on the Aggregation and Flame datasets in Fig. 2. Evidently the majority of the data have small η values, and only a few data are with great η . The comparison between ρ - δ graphs and η - δ graphs indicates that η is helpful in isolating cluster centers from non-density-peak data, and is more suitable to serve as a cluster center identification criterion than ρ .

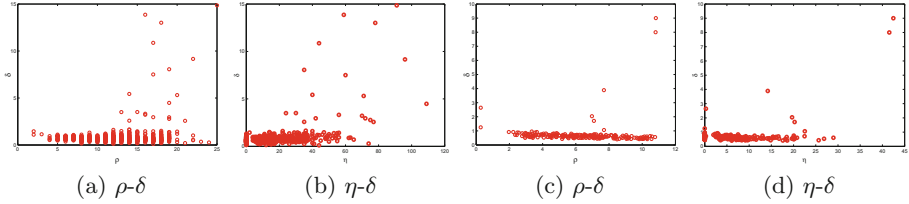


Fig. 2. The η - δ decision graphs and corresponding ρ - δ decision graphs. The left two figures belong to the Aggregation dataset, and the right two correspond to the Flame dataset.

4.2 Comparison

We now compare the proposed algorithm with some commonly used and recently proposed clustering algorithms with experiments on eight datasets, including Aggregation, Compound [18], Spiral [4], D31 [17], R15 [17], Flame [9] as well as the Wine and Iris datasets from UCI machine learning repository. Aside from the well-known k-means and DBSCAN algorithms, the normalized cuts (NCuts) [16], AP, DSets, one improved version of DSets presented in [11] and SPRG [19] are also adopted in comparison. Since our work is proposed to improve the DP algorithm, we also compare with two versions of the DP algorithm, one of which with cutoff kernel (DP-c) and the other with Gaussian kernel (DP-G). We experiment on the same eight datasets as in previous sections, and report clustering results evaluated with NMI. Except for the algorithm in [11], all these algorithms require to input one or more parameters. The k-means, SPRG and NCuts algorithms involve the number of clusters, and we set this parameter as the ground truth. As to DBSCAN which has two parameters $MinPts$ and Eps , we set $MinPts = 3$ which is selected from 2, 3, \dots , 10, and then determine Eps based on $MinPts$ [6]. The AP algorithm involves the preference value p , and the authors of [3] provide a method to obtain the range $[p_{min}, p_{max}]$ of this parameter. We sample this range and select $p = p_{min} + 9.2\xi$, where $\xi = (p_{max} - p_{min})/10$. In the DSets algorithm, $s(x, y) = \exp(-d(x, y)/\sigma)$ is used to evaluate the data similarity, and we manually select $\sigma = 10\bar{d}$ to obtain the best overall result, with \bar{d} denoting the mean pairwise distance. With the DP-c and DP-G algorithms the parameter d_c is determined by including 1.1% and 2.0% of data into the neighborhood for DP-c and DP-G, respectively. We report the clustering results of these algorithms in Table 1.

We firstly look at the comparison between the original DP algorithms DP-c, DP-G and our algorithm. On D31 and R15 datasets, both DP-c and DP-G algorithms generate very good results, and our algorithm performs as well as these two. On the Compound, Spiral, Flame, Wine and Iris datasets, our algorithm compares favorably with the two algorithms. Only on the Aggregation dataset the two DP algorithms outperform ours evidently. These comparisons demonstrate the effectiveness of our improvements to the original DP algorithm.

Table 1. Clustering results (NMI) of some algorithms.

	k-means	NCuts	DBSCAN	AP	[19]	Dsets	[11]	DP-c	DP-G	Ours
Aggregation	0.85	0.76	0.92	0.82	0.70	0.79	0.89	0.98	0.99	0.88
Compound	0.72	0.67	0.89	0.81	0.55	0.76	0.92	0.79	0.73	0.77
Spiral	0.00	0.00	0.71	0.00	0.00	0.32	0.66	0.36	1.00	1.00
D31	0.92	0.96	0.84	0.59	0.90	0.90	0.67	0.96	0.96	0.96
R15	0.96	0.99	0.87	0.74	0.94	0.86	0.91	0.98	0.99	0.99
Flame	0.39	0.44	0.83	0.57	0.30	0.50	0.90	1.00	0.41	1.00
Wine	0.43	0.36	0.38	0.39	0.87	0.77	0.43	0.61	0.71	0.74
Iris	0.74	0.76	0.75	0.79	0.73	0.64	0.56	0.66	0.66	0.86
mean	0.63	0.62	0.77	0.59	0.62	0.69	0.74	0.79	0.81	0.90

Comparatively, our algorithm is shown as the best-performing or near-best-performing one on 5 out of the 8 datasets, and our algorithm generates the best overall result. Especially on the Spiral dataset, on which k-means, NCuts and AP fail completely in clustering, our algorithm generate the perfect result. Even if our algorithm is outperformed by some algorithms on Aggregation, Compound and Wine datasets evidently, it is always among the 5 best-performing algorithms. These observations indicate that our algorithm has nice generality and performs well on various types of datasets.

5 Conclusions

An enhanced cluster center identification criterion and a new density kernel are presented to improve the DP clustering algorithm in this paper. By treating local density peaks as candidates of cluster centers, the DP algorithm uses local density and the distance to the nearest data of greater local density to represent the data and identify cluster centers. By studying the cluster center identification process, we find that local density is not very effective in strengthening the specificity of cluster centers. We introduce the concept of subordinates and present an alternative criterion to local density based on the subordinates. Furthermore, we make use of the average distance to neighboring data to evaluate the local density, in an endeavor to overcome the drawbacks of the cutoff and Gaussian kernels. Experiments show that the new criterion strengthens the specificity of cluster centers, and our algorithm performs well in comparison with some commonly used and recently proposed algorithms.

Acknowledgment. This work is supported in part by the National Natural Science Foundation of China under Grant No. 61473045, and by the Natural Science Foundation of Liaoning Province under Grant No. 20170540013 and 20170540005.

References

1. Achtert, E., Böhm, C., Kröger, P.: DeLi-Clu: boosting robustness, completeness, usability, and efficiency of hierarchical clustering by a closest pair ranking. In: Ng, W.-K., Kitsuregawa, M., Li, J., Chang, K. (eds.) PAKDD 2006. LNCS (LNAI), vol. 3918, pp. 119–128. Springer, Heidelberg (2006). https://doi.org/10.1007/11731139_16
2. Ankerst, M., Breunig, M.M., Kriegel, H.P., Sander, J.: Optics: ordering points to identify the clustering structure. In: ACM SIGMOD International Conference on Management of Data, pp. 49–60 (1999)
3. Brendan, J.F., Delbert, D.: Clustering by passing messages between data points. *Science* **315**, 972–976 (2007)
4. Chang, H., Yeung, D.Y.: Robust path-based spectral clustering. *Pattern Recogn.* **41**(1), 191–203 (2008)
5. Cheng, Y.: Mean shift, mode seeking, and clustering. *IEEE Trans. Pattern Anal. Mach. Intell.* **17**(8), 790–799 (1995)
6. Daszykowski, M., Walczak, B., Massart, D.L.: Looking for natural patterns in data: part 1. density-based approach. *Chemometr. Intell. Lab. Syst.* **56**(2), 83–92 (2001)
7. Ester, M., Kriegel, H.P., Sander, J., Xu, X.W.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: International Conference on Knowledge Discovery and Data Mining, pp. 226–231 (1996)
8. Evanno, G., Regnaut, S., Goudet, J.: Detecting the number of clusters of individuals using the software structure: a simulation study. *Mol. Ecol.* **14**(8), 2611–2620 (2005)
9. Fu, L., Medico, E.: Flame, a novel fuzzy clustering method for the analysis of dna microarray data. *BMC Bioinf.* **8**(1), 1–17 (2007)
10. Gionis, A., Mannila, H., Tsaparas, P.: Clustering aggregation. *ACM Trans. Knowl. Discov. Data* **1**(1), 1–30 (2007)
11. Hou, J., Gao, H., Li, X.: DSets-DBSCAN: a parameter-free clustering algorithm. *IEEE Trans. Image Process.* **25**(7), 3182–3193 (2016)
12. Monti, S., Tamayo, P., Mesirov, J., Golub, T.: Consensus clustering: a resampling-based method for class discovery and visualization of gene expression microarray data. *Mach. Learn.* **52**(1–2), 91–118 (2003)
13. Ng, A., Jordan, M., Weiss, Y.: On spectral clustering: analysis and an algorithm. In: Advances in Neural Information Processing Systems, pp. 849–856 (2002)
14. Pavan, M., Pelillo, M.: Dominant sets and pairwise clustering. *IEEE Trans. Pattern Anal. Mach. Intell.* **29**(1), 167–172 (2007)
15. Rodriguez, A., Laio, A.: Clustering by fast search and find of density peaks. *Science* **344**, 1492–1496 (2014)
16. Shi, J., Malik, J.: Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **22**(8), 167–172 (2000)
17. Veenman, C.J., Reinders, M., Backer, E.: A maximum variance cluster algorithm. *IEEE Trans. Pattern Anal. Mach. Intell.* **24**(9), 1273–1280 (2002)
18. Zahn, C.T.: Graph-theoretical methods for detecting and describing gestalt clusters. *IEEE Trans. Comput.* **20**(1), 68–86 (1971)
19. Zhu, X., Loy, C.C., Gong, S.: Constructing robust affinity graphs for spectral clustering. In: IEEE International Conference on Computer Vision and Pattern Recognition, pp. 1450–1457 (2014)



An Improved Weighted ELM with Hierarchical Feature Representation for Imbalanced Biomedical Datasets

Liyuan Zhang, Jiashi Zhao^(✉), Huamin Yang^(✉), Zhengang Jiang,
and Weili Shi

School of Computer Science and Technology,
Changchun University of Science and Technology,
No. 7089, Weixing Road, Changchun, China
{zhaojiashi, yhm}@cust.edu.cn

Abstract. In medical intelligent diagnosis, most of the real-world datasets have the class-imbalance problem and some strong correlation features. In this paper, a novel classification model with hierarchical feature representation is proposed to tackle small and imbalanced biomedicine datasets. The main idea of the proposed method is to integrate extreme learning machine-autoencoder (ELM-AE) into the weighted ELM (W-ELM) model. ELM-AE with norm optimization is utilized to extract more effective information from raw data, thereby forming a hierarchical and compact feature representation. Afterwards, random projections of learned feature results view as inputs of the W-ELM. An adaptive weighting scheme is designed to reduce the misclassified rate of the minority class by assigning a larger weight to minority samples. The classification performance of the proposed method is evaluated on two biomedical datasets from the UCI repository. The experimental results show that the proposed method cannot only effectively solve the class-imbalanced problem with small biomedical datasets, but also obtain a higher and more stable performance than other state-of-the-art classification methods.

Keywords: Medical intelligent diagnosis · Class imbalance data
Weighted ELM · ELM-AE

1 Introduction

The massively unbalanced nature of real-world datasets is one of major challenges in medical diagnosis application [1]. Only a small number of labeled pathological samples are available, leading to the unbalancedness between normal and abnormal. Indeed, the misdiagnosis of abnormal class is more severe than that of normal class. Therefore, the imbalanced biomedical classification has higher risk for than other fields [2]. It propels us to specifically design an effective algorithm to solve the biomedical class-imbalanced problem.

Recent years, class-imbalanced problem has drawn a significant amount of attention [3–5]. Main common methods can be grouped into two categories: data-level methods, algorithm-level methods, and hybrid methods. From the data perspective, over-

sampling and under-sampling [6] are often used to resample the dataset before training classifier. The synthetic minority oversampling technique (SMOTE) and extreme learning machine (ELM) [7] were integrated to provide an efficient solution of the imbalance classification. SMOTE creates synthetic samples to oversample the minority samples rather than mere data duplicating. This method has high learning speed and better generalization capacity. Despite the goodness, this random sampling is easy to cause the information loss. From algorithm perspective, the importance of minority samples is given full consideration. Similar to cost-sensitive learning, Zong et al [8] proposed a weighted extreme learning machine (W-ELM) method to deal with class-imbalanced problem. Samples belonging to different classes are assigned different weight values. This method boosts the accuracy of minority samples by changing its penalty factor. In general, tiny and effective features hidden in the biomedical dataset are difficult to accurately be extracted. Good feature representations can reduce the non-informative inter-class variability, whilst preserving discriminative information across classes [9]. Therefore, an effective feature representation should be introduced into medical imbalanced classification model.

The main contributions of this work are: an improved W-ELM with hierarchical feature representation has been developed for small class-imbalanced biomedical datasets; the extreme learning machine-autoencoder (ELM-AE) with norm optimization is utilized to extract more compact and meaningful features from raw data; In original W-ELM, an adaptive weighting schema is designed to adjust the weights of samples according to the imbalanced ratio, thereby reducing the misclassified rate of the minority class.

2 Brief on Weighted ELM

In this section, the preliminaries of W-ELM are briefly described. Given the sample dataset $\{(x_i, y_i)\}_{i=1}^N$, where x_i denotes an $N \times 1$ input node and y_i denotes the desired output of the i th sample. Different from original ELM, a $N \times N$ diagonal matrix $\mathbf{W} = \text{diag}(W_{11}, W_{22}, \dots, W_{NN})$ is allocated to each sample x_i . To maximize the marginal distance and minimize the weighted cumulative error, the optimization problem is mathematically written as [8]

$$\text{Minimize} : L_{D_{ELM}} = \frac{1}{2} \|\boldsymbol{\beta}\|^2 + C \frac{1}{2} \sum_{i=1}^N (W_{ii} \times \|\boldsymbol{\varepsilon}_i\|^2) \tag{1}$$

$$\text{Subject to} : h(x_i)\boldsymbol{\beta} = t_i^T - \boldsymbol{\varepsilon}_i^T \tag{2}$$

where W_{ii} is the weight of the i th training sample. $\boldsymbol{\beta}$ represents the output weight vector connecting the hidden layer and output layer. t_i is the desired output and the predicted error is expressed as $\boldsymbol{\varepsilon}_i$. $h(\cdot)$ is the feature mapping vector of the sample x_i in hidden layer. C is the penalty factor to control the balance relationship between the minimized weighted cumulative error and maximized marginal distance.

Sequentially, according to the Karush-Kuhn-Tucker (KKT) theorem [10], the following solution for W-ELM can be obtained

$$\beta_{WELM} = \begin{cases} \mathbf{H}^T (\frac{1}{C} + \mathbf{W}\mathbf{H}^T\mathbf{H})^{-1}\mathbf{W}\mathbf{T}, & \text{when } N < L \\ (\frac{1}{C} + \mathbf{W}\mathbf{H}^T\mathbf{H})^{-1}\mathbf{W}\mathbf{H}^T\mathbf{T}, & \text{when } N \geq L \end{cases} \quad (3)$$

Here \mathbf{H} denotes the output matrix of hidden-layer in ELM. L is the number of hidden node. The weight and bias of hidden layer are randomly selected and are not adjusted.

3 The Proposed Method

In this paper, inspired by hierarchical ELM (H-ELM) [11], an improved W-ELM framework (for short H-WELM) is presented for imbalanced biomedical classification. Figure 1 shows the network structure of our method. The original input is decomposed into multiple hidden layers, and the outputs of the previous layer are used as the input of the current one. By doing so, more important information can be exploited for hidden layer feature representation. The result of each layer is unchanged, without complex parameter modification, which has faster training speed.

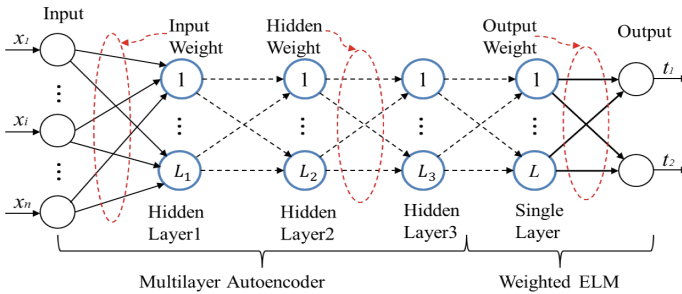


Fig. 1. The network structure of H-WELM: three-layer feature learning with two hidden-node WELM feature classification

3.1 Hierarchical Feature Representation

ELM-AE based ℓ_1 -norm optimization is used to obtain more compact and sparse hidden information by searching the path back from a random space. A fast iterative shrinkage-thresholding algorithm (FISTA) [12] is employed to obtain the output weight β , the optimization model of ELM-AE is

$$Output_{\beta} = \arg \min_{\beta} \left\{ \|\beta\|_{\ell_1} + \|\mathbf{H}\beta - \mathbf{X}\|^2 \right\} \quad (4)$$

where β represents the output weights. $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_N]$ stands for the input vector; $\mathbf{H} = [\mathbf{h}_1, \dots, \mathbf{h}_N]$ represents the random mapping output of hidden layer. Mathematically, the output of the i th hidden layer ($i \in (1, \dots, K)$) can be represented as

$$H_i = G(H_{i-1} \cdot \beta) \tag{5}$$

$$G(\mathbf{a}, b, \mathbf{x}) = \frac{1}{1 + \exp(-(\mathbf{a} \cdot \mathbf{x} + b))} \tag{6}$$

Here, sigmoid function is selected as the hidden layer node function $G(\cdot)$.

3.2 Imbalanced Data Classification via W-ELM

After achieving more compact feature set, the original W-ELM is performed for final decision making. Assume that N^+ represents the number of the minority samples and N^- represents the number of the majority samples. The definition of imbalanced ratio is formally described as $IR = N^+ / N^-$. When IR is smaller than one, an uneven distribution of samples in the training set will occur. For this, an adaptive weighting selection method is designed according to the imbalanced ratio. The weight of W_{ii} for each sample is defined as follows

$$W_{ii} = \begin{cases} W^- = \frac{N^+}{N^-}, & N^- > \frac{N}{2} \\ W^+ = 1, & N^+ \leq \frac{N}{2} \end{cases} \tag{7}$$

where $N^- + N^+ = N$. W^- denotes the weight associated with the majority samples. While W^+ is the weight associated with the minority samples. The minority samples will be given larger weight. For a sample x , the output function of W-ELM model is expressed as

$$f(x) \begin{cases} \text{sign } h(x) \mathbf{H}^T (\frac{1}{C} + \mathbf{W} \mathbf{H}^T \mathbf{H})^{-1} \mathbf{W} \mathbf{T}, & \text{when } N < L \\ \text{sign } h(x) (\frac{1}{C} + \mathbf{W} \mathbf{H}^T \mathbf{H})^{-1} \mathbf{W} \mathbf{H}^T \mathbf{T}, & \text{when } N \geq L \end{cases} \tag{8}$$

4 Experimental Results and Discussion

In this section, to demonstrate the effectiveness and superiority of the proposed algorithm, we conduct experiments on two real-world imbalanced biomedical datasets derived from the UCI machine learning repository [13]. Moreover, we compare the classification performance of the H-WELM with SMOTE-ELM [7] and W-ELM [8]. Our experimental computer configurations are listed as follows: Intel(R) dual-core, 2.93 GHz, 8 GB RAM with Windows 7 Operating System. All algorithms are implemented in MATLAB 2014a.

Datasets Description Detailed information about two datasets is summarized in Table 1. The breast cancer diagnostic dataset provides information on the discrimination

of benign and malignant. Each instance has one ID number, 30 real value variables, and one diagnosis label. The BUPA liver disorders dataset consists of six features plus a class label. The features are diagnostic markers that are representative of liver disease. It is noticeable that all datasets should be normalized to facilitate processing. For binary classification, the labels of majority class and minority class are marked as 0 and 1, respectively.

Table 1. Description of imbalanced datasets

Datasets	Instances	Attributes	Imbalance ratio
Breast Cancer	569	32	212:357
Liver Disorders	345	7	145:200

Performance Evaluation Criteria There are several commonly considered evaluation criteria that can be used in imbalance classification task. They are sensitivity, specificity, geometric mean (G-mean), receiver operator characteristic (ROC), and area under the curve of ROC (AUC), which can be defined as

$$\text{G-mean} = \sqrt{\text{TPR} \cdot \text{TNR}}, \text{AUC} = \frac{1 + \text{TP} - \text{FP}}{2} \quad (9)$$

$$\text{Sensitivity} = \text{TPR} = \left(\frac{\text{TP}}{\text{TP} + \text{FN}} \right), \text{Specificity} = \text{TNR} = \left(\frac{\text{TN}}{\text{FP} + \text{TN}} \right) \quad (10)$$

where, TP and TN are the number of correctly classified examples belonging to the positive and negative classes, FP and FN are the number of misclassified examples belonging to the negative and positive classes, respectively.

The Analysis of Results In order to compare the classification performance of H-WELM method with W-ELM and SMOTE-ELM, we test these algorithms on the two biomedical datasets (see Fig. 2). For fair comparison, the parameters of these ELM approaches are set to be equal to that of our method. Considering the weight of ELM models was randomly chosen, all methods were run ten times, the final experimental result was from the average of ten results. Additionally, for related parameters, the activation function of the hidden layer is set as the sigmoid activation function. For smaller training samples, three hidden layers is appropriate for feature extraction. The weight and bias of hidden node are randomly produced range in $[-1, 1]$. The regularization parameter $C = 2^2$ will make the learning performance better.

It is obvious from Fig. 2 that the H-WELM method can get relatively higher performance than those of other two methods. The H-WELM has a remarkable improvement against the W-ELM. The original W-ELM directly assigns the fixed weight for each sample to deal with imbalanced data classification. It is difficult to get the optimal result for all datasets. The key reason is that the outlier or noisy sample not to be processed in W-ELM. Moreover, SMOTE-ELM applies the undersampling of the

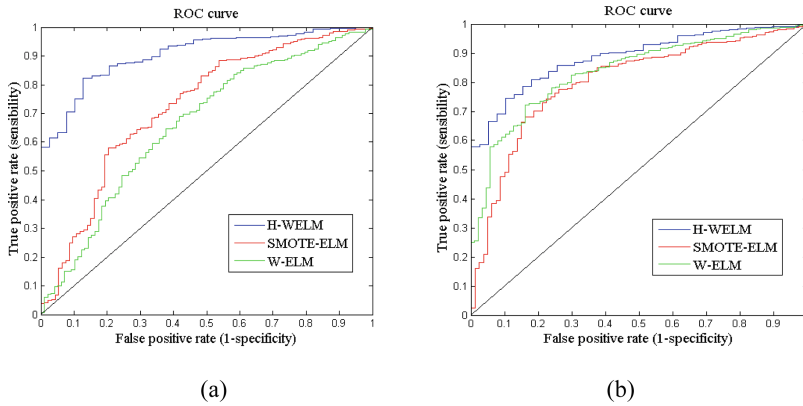


Fig. 2. Comparison of ROC curves using SMOTE-ELM, W-ELM, and H-WELM on (a) liver disorders dataset and (b) breast cancer dataset.

majority and the oversampling of the minority samples to balance the sample data. If SMOTE blindly broadens the minority class without regard to the distribution of the majority class, the classification performance of SMOTE-ELM is lower. But the H-WELM employs an unsupervised learning process to represent compact features information. Considering the imbalance ratio into the weighting schema can adaptively assign the weight of majority samples to better solve class-imbalance.

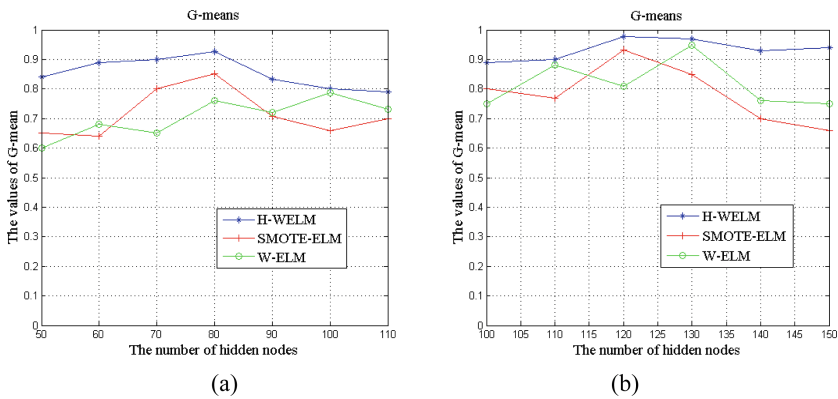


Fig. 3. G-means with different numbers of hidden nodes on (a) liver disorders dataset and (b) breast cancer dataset.

To strengthen the reliability of the H-WELM method, G-means along with different numbers of hidden node are shown in Fig. 3. In general, the optimal number of hidden nodes may be related with the number of samples. From Fig. 3, G-means of W-ELM and H-ELM have a slight fluctuation as changing hidden nodes, while all G-means of H-WELM has not significantly changed and are nearly optimal. The whole performance of

our method is not much sensitive to the number of hidden nodes and more stable. At last, Table 2 presents more comparison results on two uneven biomedical datasets with respect to G-mean, AUC, sensitivity, and specificity.

Table 2. Comparative performance results on two datasets

Datasets	Methods	G-mean	AUC	Sensitivity	Specificity
Breast cancer	W-ELM	0.9316	0.9367	0.8831	0.8680
	SMOTE-ELM	0.9487	0.9247	0.9105	0.9001
	H-WELM	0.9761	0.9461	0.9588	0.9529
Liver disorders	W-ELM	0.7882	0.7199	0.8028	0.6213
	SMOTE-ELM	0.8503	0.8446	0.8384	0.7231
	H-WELM	0.9260	0.9343	0.9068	0.8576

From Table 2, we observe the following facts. Firstly, the H-WELM has larger AUC against W-ELM and SMOTE-ELM because of its adaptive weighted strategy. The negative effect of data preprocessing is avoided. Secondly, the comparison results of G-means show that the generalization performance of W-ELM has indeed been enhanced by providing more robust features extraction from raw imbalanced data. Thirdly, the H-WELM always acquires higher sensitivity and specificity values. It is highly desirable to eliminate these false positives as much as possible while retaining the true positives. The missed diagnosis and the misdiagnosis rate have been reduced. The obtained results of two datasets are quite promising, and further confirm the generality and capability of the H-WELM.

5 Conclusion

In this paper, an improved weighted ELM with hierarchical feature representation has been proposed for imbalanced biomedical classification. Different from original W-ELM, the proposed method employs hierarchical ELM-AE to extract more effective features before classifying. The generated informative feature set is learned by W-ELM. During the training process, adaptive weight values are assigned to reduce the misclassified rate of minority samples. Comparative experimental results prove that the proposed method demonstrates better the classification performance. How to automatically adjust the hyper-parameters of the hidden layer is our future research task.

Acknowledgments. This work is supported by the Science & Technology Development Program of Jilin Province, China (Nos. 20150307030GX, 2015Y059 and 20160204048GX), and by the International Science and Technology Cooperation Program of China under Grant (No. 2015DFA11180), National Key Research and Development Program of China (No. 2017YFC0108303), and Science Foundation for Young Scholars of Changchun University of Science and Technology (No. XQNJJ-2016-08).

References

1. Krawczyk, B.: Learning from imbalanced data: open challenges and future directions. *Prog. Artif. Intell.* **5**(4), 221–232 (2016)
2. Rahman, M.M., Davis, D.N.: Addressing the class imbalance problem in medical datasets. *Int. J. Mach. Learn. Comput.* **3**(2), 224–228 (2013)
3. Krawczyk, B., Galar, M., Jelen, Ł., Herrera, F.: Evolutionary undersampling boosting for imbalanced classification of breast cancer malignancy. *Appl. Soft Comput.* **38**, 714–726 (2016)
4. Ali, S., Majid, A., Javed, S.G., et al.: Can-CSC-GBE: developing cost-sensitive classifier with gentleboost ensemble for breast cancer classification using protein amino acids and imbalanced data. *Comput. Biol. Med.* **73**, 38–46 (2016)
5. Ren, F., Cao, P., Li, W., et al.: Ensemble based adaptive over-sampling method for imbalanced data learning in computer aided detection of microaneurysm. *Comput. Med. Imaging Graph.* **55**, 54–67 (2017)
6. Yap, B.W., Rani, K.A., Rahman, H.A.A., Fong, S., Khairudin, Z., Abdullah, N.N.: An application of oversampling, undersampling, bagging and boosting in handling imbalanced datasets. In: Herawan, T., Deris, M.M., Abawajy, J. (eds.) *Proceedings of the First International Conference on Advanced Data and Information Engineering (DaEng-2013)*. LNEE, vol. 285, pp. 13–22. Springer, Singapore (2014). https://doi.org/10.1007/978-981-4585-18-7_2
7. Gong, C.L., Gu, L.X.: A novel SMOTE-based classification approach to online data imbalance problem. *Math. Probl. Eng.*, 1–14 (2016)
8. Zong, W., Huang, G.B., Chen, Y.: Weighted extreme learning machine for imbalance learning. *Neurocomputing* **101**, 229–242 (2013)
9. Sani, S., Massie, S., Wiratunga, N., Cooper, K.: Learning deep and shallow features for human activity recognition. In: Li, G., Ge, Y., Zhang, Z., Jin, Z., Blumenstein, M. (eds.) *KSEM 2017. LNCS (LNAI)*, vol. 10412, pp. 469–482. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-63558-3_40
10. Huang, G., Huang, G.B., Song, S., et al.: Trends in extreme learning machines: a review. *Neural Netw.* **61**, 32–48 (2015)
11. Tang, J.X., Deng, C.W., Huang, G.B.: Extreme learning machine for multilayer perceptron. *IEEE Trans. Neural Netw. Learn. Syst.* **27**(4), 809–821 (2016)
12. Beck, A., Teboulle, M.: A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Sci.* **2**(1), 183–202 (2009)
13. UCI Machine Learning Repository. <http://archive.ics.uci.edu/ml/datasets>

Recommendation Algorithms and Systems



SERL: Semantic-Path Biased Representation Learning of Heterogeneous Information Network

Haining Tan^{1,2} , Weiqiang Tang^{1,2} , Xinxin Fan^{1,2} , Quanliang Jing^{1,2} ,
and Jingping Bi^{1,2}  

¹ Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
{tanhaining, tangweiqiang, fanxinxin, jingquanliang, bjp}@ict.ac.cn

² University of Chinese Academy of Sciences, Beijing, China

Abstract. The goal of network representation learning is to embed each vertex in a network into a low-dimensional vector space. Existing network representation learning methods can be classified into two categories: *homogeneous* models that learn the representation of vertexes in a homogeneous information network, and *heterogeneous* models that learn the representation of vertexes in a heterogeneous information network. In this paper, we study the problem of representation learning of heterogeneous information networks which recently attracts numerous researchers' attention. Specifically, the existence of multiple types of nodes and links makes this work more challenging. We develop a scalable representation learning models, namely *SERL*. The *SERL* method formalizes the way to fuse different semantic paths during the random walk procedure when exploring the neighborhood of corresponding node and then leverages a heterogeneous skip-gram model to perform node embeddings. Extensive experiments show that *SERL* is able to outperform state-of-the-art learning models in various heterogenous network analysis tasks, such as node classification, similarity search and visualization.

Keywords: Heterogeneous information network
Representation learning · Semantic path · Classification
Similarity search

1 Introduction

Heterogeneous information networks (HINs) are the logical networks including multiple types of objects and links denoting different relations, such as social media network, world wide web (WWW), bibliographic network [12]. As HIN contains more knowledge we needed to explore the real world, HIN's analysis gradually becomes the focus both in research and industry field. HIN's analysis often involves proximity search and prediction tasks over nodes or edges, e.g., node classification, link predication, node clustering [13, 16]. For the purpose of

getting good performance of these tasks, an appropriate representation method that preserves both the HINs’ physical structures and relationships among nodes is much needed to serve as input features to supervised machine learning algorithms. This is a preprocessing step for data mining and knowledge discovery, which is called feature engineering. Usually, this process, carefully designed by domain-specific experts according to their knowledge and experiences, is time consuming and expensive. To solve this problem, many researchers have shown a great deal of interests into networks representation learning that aims to embed a network into a low-dimensional space and represents each node as a low-dimensional feature vector for supervised learning.

In recent years, there are a few ways that aim to get great representations of networks like DeepWalk [10], LINE [14], node2vec [7] and metapath2vec [5]. However, they’re either initially not suitable for heterogenous environment or carefully designed for discriminating different semantic paths. Take Fig. 2 as an example, there are multiple paths connecting two objects a_1 and a_3 , like paths(① ~ ④). For ① and ④, ①($a_1 \rightarrow p_1 \rightarrow a_3$) can be seen as an instance of $\mathcal{P}_{①} : A \rightarrow P \rightarrow A$ while ④($a_1 \rightarrow o_1 \rightarrow a_3$) is an instance of meta path $\mathcal{P}_{④} : A \rightarrow O \rightarrow A$. As we can see, they convey different semantic relationship of two objects, which we should treat them separately when we analyze the HIN. But many methods use one specific meta path \mathcal{P}_i to guide the procedure of random walk, like $a_1 \rightarrow a_3 \rightarrow a_4$. When one edge is removed, the path is truncated like Fig. 1 shows. Thus we need to consider how to merge different meta paths to keep the proximity of a_1 , a_3 and a_4 like introduced in Fig. 1. To tackle the aforementioned challenges, we present a novel idea on network representation learning, termed Semantic path biased Representation Learning of HINs, which jointly consider both the structure as well as the semantic information. Our contributions are as follows:

- Investigate and formalize representation learning problems in HINs, a new but increasing important issue due to the proliferation of linked data and its abundant applications.
- Design effective and efficient network embedding framework, Semantic path biased Representation Learning of HIN, to merge different semantic paths during the learning procedure.
- Through extensive experiments on real world heterogeneous information networks, illustrating the efficacy and scalability of the designed representation learning method in mining HINs’ tasks and the importance of discriminating different semantic paths.

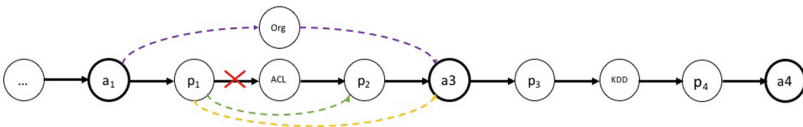


Fig. 1. Example of random walk

2 Related Work

Initially, networks can be represented by adjacency matrices. However, due to the sparsity and the high-dimensional of this representation method, many classic methods such as laplacian eigenmap [1] have been proposed to get low-dimensional representation. These ways of reducing dimensions can work fine with small size networks but cannot applied to large networks. In DeepWalk [10], the authors develop an algorithm(DeepWalk) that learns social representations of a graph’s vertices, by modeling a stream of truncated random walks, inspired by word2vec [8]. Instead, node2vec [7], a semi-supervised algorithm for scalable feature learning in networks, uses a 2^{nd} order random walk approach to generate (sample) network neighborhoods for nodes which extends DeepWalk. In [14], the authors propose a novel model that preserves both the first-order and second-order proximities. GraRep [2] introduces a latent representations of vertices on graphs, which can capture global structural information associated with the graph. SDNE [17] uses autoencoder to capture first-order and second-order network structures and learn user representation. However, aforementioned methods just can address homogeneous networks. When these methods come to HINs, the efficiency isn’t quite enough for some network analysis tasks. Despite this, there are a few works aim to conquer problems when represent HINs. In [5], the author develops heterogeneous networks embedding frameworks, *metapath2vec&metapath2vec++*, preserving both structure and semantic correlations. In [3], the authors present HNE, mapping different heterogeneous objects into a unified latent space based on DNN. HNE is almost the first try in using deep architecture on representing HINs. In [11], the authors present an effective map method in translating HIN embedding to homogenous embedding problem. The authors first get random walks based on given meta-path, similar way used in [5]. Once the sequence has been constructed, they further remove the nodes with different types. In this way, they finally obtain a homogenous node

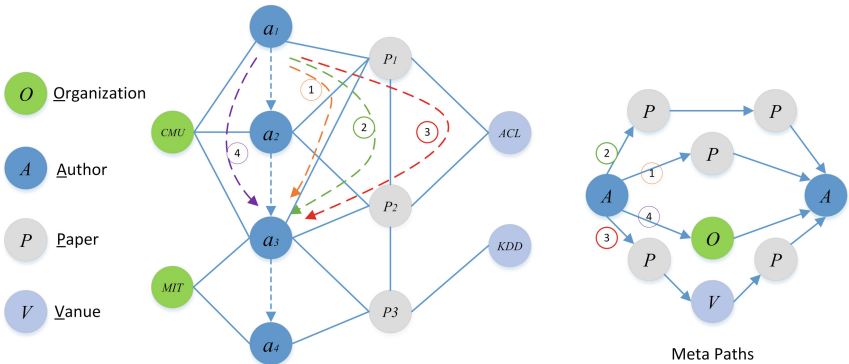


Fig. 2. Bibliographic network

sequence which can be easily handled. [4] proposes a task-guided and path-augmented HIN embedding framework and also uses experiments to demonstrate the usefulness of selecting specific meta-path in HIN embedding while facing a specific analysis task. HIN2VEC [6] adopts a single-hidden-layer feedforward neural network model, which is a binary classifier that capture different relationships between typed nodes. SHINE [18] proposes a novel and flexible end-to-end signed HIN embedding method, which utilizes multiple deep autoencoder to map each user into a low-dimension feature space, while preserving the network structure.

3 Problem Preliminaries

Definition 1 *Heterogeneous Information Network.* An information network is a directed graph $G = (V, E, \Phi, \Upsilon)$, where V is the set of nodes, $E \subseteq V \times V$ is the set of edges denoting by nodes in V . Φ is the set of vertex types, which means each node in V is mapped to a particular node type in T , can be formulated as mapping function $\Phi : V \rightarrow T$. Υ is the set of link types, which means each edges in E is mapped to a particular edge type in R , can be formulated as mapping function $\Upsilon : E \rightarrow R$. When $|T| > 1$ or $|R| > 1$, the network is called heterogeneous information network; otherwise, it is called homogeneous information network.

Definition 2 *Meta-path.* A meta-path \mathcal{P} is defined on network schema $\phi = (T, R)$ and is denoted as a path in the form of $T_1 \xrightarrow{R_1} T_2 \xrightarrow{R_2} \dots \xrightarrow{R_n} T_{n+1}$, which describes a composite relation $R = R_1 \odot R_2 \dots \odot R_n$ between objects T_1 and T_{n+1} , where \odot denotes the composition operator on relations.

Definition 3 *Representation Learning on HIN.* Given a large HIN $G = (V, E, \Phi, \Upsilon)$, The problem of representation learning on HIN aims to represent each vertex $v \in V$ into a low-dimensions space \mathbb{R}^d , i.e., learning a function $f_G : V \rightarrow \mathbb{R}^d$, where $d \leq |V|$. In the \mathbb{R}^d , both the structure and the latent representations between vertices are preserved.

4 The HIN’s Representation Learning Framework

Here we present a general framework of getting latent representation on HINs. We need to maximize the network probability in consideration of multiple nodes and edges. Next, we will introduce the skip-gram model and the semantic path biased random walks separately.

4.1 The Skip-Gram Model for HINs

Skip-gram model is an effective method for learning high-quality distributed vector representations that capture a large number of precise syntactic and semantic

word relationships which is first used in word2vec. Here, we map the word-context concept in text corpus into a network.

Usually, given a homogeneous network $G = (V, E)$, the objective is to maximize the network probability in terms of local structures, that is:

$$\arg \max_{\theta} \prod_{v \in V} \prod_{n \in N_v} p(n|v, \theta) \tag{1}$$

where N_v is the neighborhood of the node v in the network G and $p(n|v, \theta)$ is the conditional probability of having a context node n given a node v (Fig. 3).

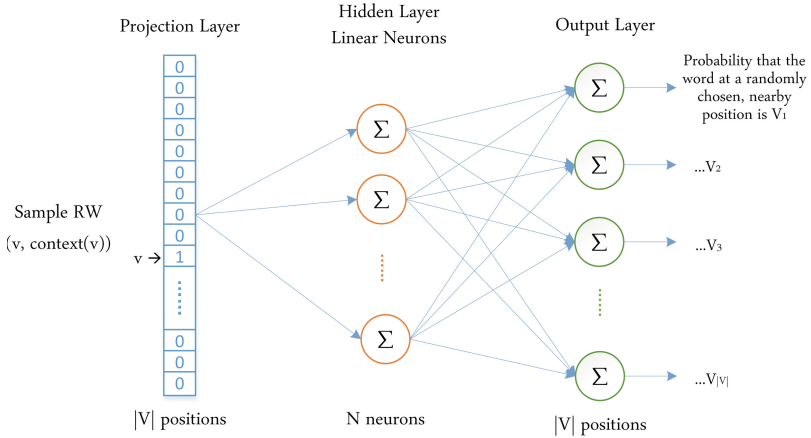


Fig. 3. Semantic-path biased framework for HINs

Given a HIN $G = (V, E, \Phi, \Upsilon)$ with $|T_V| > 1$, for $v \in V$, we should maximize the probability of having the heterogeneous context RW_v :

$$\ell(n, v, \theta, \mu) = \arg \max_{\theta} \sum_{v \in V} \sum_{n \in RW_{(v)}} \log p(n|v, \theta, \mu) \tag{2}$$

where $RW_{(v)}$ denotes v 's neighborhood creating by given-length random walks of node v and $p(n|v, \theta, \mu)$ is common defined as a hierarchical softmax function, that is:

$$p(n|v, \theta, \mu) = \frac{e^{R_n^T R_v}}{\sum_{\rho \in V} e^{R_\rho^T R_v}} \tag{3}$$

where R is the embedding vectors for nodes in V and R_v is the v^{th} row of R that is the representation of node v . Take the academic network Fig. 2. as an example, the random-walk neighborhood of one author node a_3 can be structurally close to other authors (e.g. a_2 & a_4), papers (e.g. p_1 , p_2 & p_3), venues (e.g. ACL &

KDD), organizations (e.g. CMU & MIT). θ is the auxiliary vector of skip-gram model.

To achieve efficient optimization, Mikolov et al. introduces negative sampling, an alternative way to $\log P(n|v, \theta, \mu)$, the objective can be described as follows:

$$\mathcal{L}(n, v, \theta, \mu) = \log \sigma(R_n^T R_v) + \sum_{i=1}^k \mathbb{E}_{v_i \sim P_n(v)} [\sigma(-R_{v_i}^T R_v)] \quad (4)$$

where $\sigma(x) = 1/(1 + e^x)$. Thus, the task is to distinguish the target vector v_n from the noise distribution $P_n(v)$ using logistic regression. The parameter k is the negative samples for each node sample v_n . [9] indicate that values of k in the range 5–20 are useful for small training datasets, while for large datasets the k can be small as 2–5. The noise distribution $P_n(v)$ is a free parameter for negative sampling. Here, we set $P_n(v) \propto d_v^{3/4}$, as proposed in [9], where d_v is the out-degree of vertex v in HINs.

Next, we use stochastic gradient descent (SGD) to optimize Eq. (4). In each iteration, R_n and R_v can be updated following adaptive learning rate formulated in Eq. (5) and Eq. (6):

$$\frac{\partial(\mathcal{L}(n, v, \theta, \mu))}{\partial(R_n^i)} = [\sigma(\mathbb{I}_{(n, v, \theta, \mu)}(v_i) - R_n^{i T} R_v)] R_v \quad (5)$$

$$\frac{\partial(\mathcal{L}(n, v, \theta, \mu))}{\partial(R_v)} = \sum_{i=0}^{|R_n|} [\sigma(\mathbb{I}_{(n, v, \theta, \mu)}(v_i) - R_n^{i T} R_v)] R_n^i \quad (6)$$

4.2 Semantic-Path Biased Random Walk

In DeepWalk [10], the random walk generator takes a graph G and samples uniformly a random vertex v_i as the root of the random walk RW_{v_i} . A walk samples uniformly from the neighbors of the last vertex visited until the maximum length l is reached. When it comes to HINs, as there are multi-typed nodes and edges, we should take the sample walk process carefully. As shown in Fig. 2, there are many meta paths connecting two objects for a given $G = (V, E, \Phi, \Upsilon)$ with different semantic.

Generally speaking, there is no constraint on the length of the meta path which means the meta path can grow exponentially with their length. However, as pointed out in [13], shorter meta paths are more informative than the longer one, because longer paths connect remote objects (which are less related semantically). Therefore, we use a truncated estimation in our random walk procedure, which only considers meta paths up to a length threshold l . In this paper, we consider learning such linear weighting schemes over all relation paths of bounded length l . For small l (e.g., $l \leq 4$), we can easily generate the meta-path set:

$$\zeta = \bigcup \mathcal{P}_{(v_s, \dots, v_t | l)}, \forall (v_i)_{\mathcal{P}} \in V, |\mathcal{P}| < l. \quad (7)$$

In order to effectively transform the structure of a network into skip-gram, we should take different meta paths with discriminative weights. For $\forall \mathcal{P} \in \zeta$, there is a specific weight $\mu_{\mathcal{P}}$ which introduces a semantic-path bias. Thus, an instance of random walk $RW_{v_s} = (v_s, \dots, v_t), v_i \in V$ can be seen as the concatenation of instances of $\mathcal{P} \in \zeta$, can be formulated as the following equation:

$$RW_{v_s} = (v_s, \dots, v_t) = (\mathcal{P}_{inst}^1, \mathcal{P}_{inst}^2, \dots, \mathcal{P}_{inst}^n), v_i \in V, \mathcal{P}^i \in \zeta \quad (8)$$

Thus, there are two steps to finish the random walk procedure. Firstly, we should choose which meta path \mathcal{P} should we use to create the instance, the probability $p(\mathcal{P}^n | \mathcal{P}^{n-1})$ can be computed as follows:

$$p(\mathcal{P}^n | \mathcal{P}^{n-1}) = \frac{|\mathcal{P}_{inst}^n| \mu_{\mathcal{P}^n}}{\sum_{i=1}^{|\zeta|} |\mathcal{P}_{inst}^i| \mu_{\mathcal{P}^i}}, 1 \leq n \leq |\zeta| \quad (9)$$

Secondly, we should choose which objects to choose following the chosen meta path $\mathcal{P}: V_1 \xrightarrow{R_1} V_2 \xrightarrow{R_2} \dots V_t \xrightarrow{R_t} V_{t+1} \dots \xrightarrow{R_{|\mathcal{P}|-1}} V_{|\mathcal{P}|}$, the probability $p(v_{i+1} | v_i)$ can be formulated as follows:

$$p(v^{i+1} | v_t^i, \mathcal{P}) = \begin{cases} \frac{1}{|N_{t+1}(v_t^i)|}, & (v^{i+1}, v^i) \in E, \Phi(v^{i+1}) = t + 1 \\ 0, & (v^{i+1}, v^i) \in E, \Phi(v^{i+1}) \neq t + 1 \\ 0, & (v^{i+1}, v^i) \notin E \end{cases} \quad (10)$$

where $v_t^i \in V_t$ and $N_{t+1}(v_t^i)$ describes the V_{t+1} type of neighborhood of node v_t^i .

In this paper, we apply semantic-path constrained random walks for several advantages. First, it is computationally efficient for both space and time complexity. The space complexity of storing the directed neighbors of a node to select the next node in a random walk is $O(|V|)$; the time complexity is $O(|V| \cdot l^D)$ where l is the length threshold and D is the vertices' average degree in the HINs which is linear to the size of the HIN. What's more, using random walks to generate the training data is scalable for large-scale networks. The pseudo code of *SERL* is listed in Algorithm 1.

5 Experiment Results

In this section, we provide an overview of the datasets and methods used for experiments and evaluate the effectiveness of our method on multiple mining tasks in real-life networks. We further evaluate the effect of several important parameters on the framework's performance.

5.1 Experiment Setup

Table 1 gives us an overview of the HIN datasets we use in our experiments.

AMiner dataset introduces an academic social network [15]. The content of this data includes paper information, paper citation and author information. Here, we use data of 59 conferences in 8 research fields to form a HIN network.

DBLP is a bibliographic information network which is frequently used in the study of HIN. Our dataset is constructed and used by Sun et al. and it is a subset of DBLP. It covers 20 venues in four areas¹, their 14475 authors and corresponding 14376 papers [13].

Algorithm 1. Semantic-path biased representation learning

Input:

HIN: $G = (V, E, T)$; Meta paths: ζ ; Walks per node: w ; Walk length: l ;
 Meta path weights $\mu = \{\mu_1, \mu_2, \dots, \mu_n\}$; Embedding dimensions: d ;

Output:

the latent node embedding R

```

1: initialize  $R$ 
2: for  $i = 1$  to  $w$  do
3:   for  $v \in V$  do
4:      $SRW = \text{SemanticBiasedRandomwalk}(G, v, \zeta, \mu, l)$ 
5:      $R = \text{HeterogenousSkipGram}(R, k, v, SRW)$ 
6: return  $R$ 
7: SemanticBiasedRandomwalk( $G, v, \zeta, \mu, l$ ) :
8:  $SRW[1] = v$ 
9: for  $i = 2$  to  $l$  do
10:  Select  $v_i$  based on semantic weights according to Eqn. (10)
11:   $SRW[i] = v_i$ 
12: return  $SRW$ 
13: HeterogenousSkipGram( $R, k, v, SRW$ ) :
14: for  $v_n \in SRW$  do
15:  for  $v \in SRW \cup \text{NEG}^{v_n}(SRW)$  do
16:     $g = \eta(Is_{RW}(v) - \sigma(R_n^T R_v))$ ,  $\eta$  is the learning rate
17:     $e = e + gR_v$ 
18:     $R_v = R_v + gR_{v_n}$ , according to Eqn. (5)
19:     $R_{v_n} = R_{v_n} + e$ , according to Eqn. (6)
20: return  $R$ 

```

Table 1. Statistics of the HINs used in our experiments

Name	Vertices	Edges	Types	Classes
AMiner	76014	165925	4	8 (author) & 8 (venue)
DBLP	138957	336641	3	8 (author) & 4 (venue)

Baselines. We compare our framework with several recent representation learning methods:

- (1) **DeepWalk**[10]: DeepWalk generates random walks of fixed length from all the vertices of a graph. Then, DeepWalk use hierarchical softmax for Skip-gram model optimization.

¹ databases, data mining, artificial intelligence and information retrieval.

- (2) **LINE**[14]: LINE is designed for preserving both first-order and second-order proximities and we use both methods for comparison.
- (3) **metapath2vec**[5]: This work generates random walks satisfying specific meta-path and then leverages skip-gram model to learn the embeddings.

Parameter Settings. Here we discuss the parameter settings for our model and baseline models. Since DeepWalk and LINE are initially designed for homogeneous network, we treat typed vertices as homogeneous objects when sampling.

For all methods, the number of walks per node w is 100; the walk length l is 100; the vector dimension d is 128. As our work needs the meta paths, we set the length threshold of meta path used to 4 and we use grid search to find the suitable μ . For AMiner, the meta paths set ζ satisfied can be formulated as [$A \rightarrow P \rightarrow A$ ”, $A \rightarrow O \rightarrow A$ ”, $A \rightarrow P \rightarrow P \rightarrow A$ ”, $A \rightarrow P \rightarrow V \rightarrow P \rightarrow A$ ”]; for DBLP, the meta paths set ζ satisfied can be formulated as [$A \rightarrow P \rightarrow A$ ”, $A \rightarrow P \rightarrow V \rightarrow P \rightarrow A$ ”].

5.2 Multi-class Classification

In multi-class classification, each vertex is assigned one or multiple labels. After learning the representation of vertices, We first use partial labeled vertices to train a SVM classifier, then apply the classifier to the rest data. The results of Accuracy and Macro-F1 are presented in table. As we can see, semantic-path biased methods outperforms all baselines on both datasets. For example, our work achieves gains of 2% to 20% on accuracy on two datasets. In this task, we just consider the multiple-class classification results of authors based on the labels according to Google Scholar² (Table 2).

Table 2. Results of multiple-class classification on AMiner

Method(%)	5%	10%	15%	20%	25%	30%	35%	40%	45%	50%
<i>DeepWalk</i>	66.53	72.22	72.34	74.03	73.80	74.13	74.58	75.86	76.14	76.12
<i>Node2Vec</i>	66.09	74.35	75.04	75.63	76.23	76.78	77.32	76.65	77.86	78.24
<i>LINE</i>	73.32	78.76	78.21	78.31	78.94	78.32	79.03	78.65	79.37	79.85
<i>metapath2vec</i>	75.89	81.02	86.69	88.03	88.76	89.43	91.34	91.87	93.42	93.78
<i>SERL*</i>	79.03	85.93	89.83	90.65	92.15	92.87	92.96	93.45	94.01	94.32

5.3 Case Study: Similarity Search

In similarity search, we run our algorithm to generate a representation vector for each vertex in AMiner, which is used as a feature representation for

² https://scholar.google.com/citations?view_op=top_venues&hl=en&vq=eng.

Accessed on February, 2017.

similarity search. In this part, we just consider the 8 research areas³, the 59 venues belongs to. We compute the similarity scores between two venues according to the product of two vectors($S = R_{v_1}R_{v_2}^T$) and then rank them. Table 4 lists the top 4 similar results for querying the 8 leading conferences in 8 areas. More surprisingly, we find that in most cases, the top three results cover conferences with similar research topic, such as INFOCOM to SIGCOMM, VLDB to SIGMOD (Table 3).

Table 3. Results of multiple-class classification on DBLP

Method(%)	5%	10%	15%	20%	25%	30%	35%	40%	45%	50%
<i>DeepWalk</i>	75.70	80.80	82.49	83.88	84.83	85.71	86.58	86.90	86.93	87.09
<i>Node2Vec</i>	77.63	82.35	82.89	84.51	85.47	86.72	86.95	87.05	88.09	88.72
<i>LINE</i>	80.43	83.55	84.56	84.98	86.12	87.22	87.79	88.09	88.12	88.56
<i>metapath2vec</i>	82.14	86.02	87.04	87.96	88.47	88.66	88.91	88.90	89.02	89.13
<i>SERL*</i>	85.20	88.97	89.99	90.78	91.42	91.65	92.13	92.42	92.42	92.51

Table 4. Case study of similarity search in AMiner data

Rank (%)	<i>ISCA</i>	<i>SODA</i>	<i>SIGGRAPH</i>	<i>SIGCOMM</i>	<i>UbiComp</i>	<i>ACL</i>	<i>CVPR</i>	<i>SIGMOD</i>
0	<i>ISCA</i>	<i>SODA</i>	<i>SIGGRAPH</i>	<i>SIGCOMM</i>	<i>UbiComp</i>	<i>ACL</i>	<i>CVPR</i>	<i>SIGMOD</i>
1	<i>ASPLOS</i>	<i>FOCS</i>	<i>VAST</i>	<i>INFOCOM</i>	<i>INTERACT</i>	<i>EMNLP</i>	<i>ECCV</i>	<i>VLDB</i>
2	<i>MICRO</i>	<i>STOC</i>	<i>Web3D</i>	<i>GLOBECOM</i>	<i>CHI</i>	<i>COLING</i>	<i>ECCV</i>	<i>SIGIR</i>
3	<i>HPCA</i>	<i>ICALP</i>	<i>VRST</i>	<i>ICC</i>	<i>IUI</i>	<i>EMNLP</i>	<i>IJNLP</i>	<i>ICDE</i>

5.4 Case Study: Visualization

Another way of assessing the quality of the vertex representations is through visualization. We conduct visualization experiments by following (Tang et al. 2015) to compare the performance of our work with DeepWalk, LINE and meta-path2vec, which mapped all vectors into a 2-dimension space. The results are presented in Fig. 4.

We can see that the visualization of the vectors from DeepWalk, LINE and meta-path2vec has unclear boundaries and diffuse clusters. Our work *SERL* is better.

³ 1. Computational Linguistics, 2. Computer Graphics, 3. Computer Networks & Wireless Communication, 4. Computer Vision & Pattern Recognition, 5. Computing Systems, 6. Databases & Information Systems, 7. Human Computer Interaction, and 8. Theoretical Computer Science.

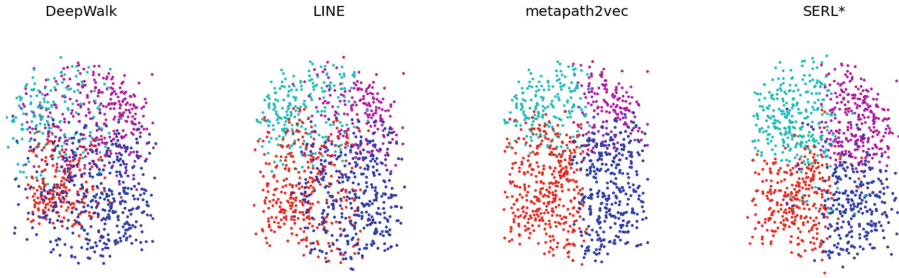


Fig. 4. Visualization of the DBLP network. 1000 authors of DBLP are mapped to the 2-D space using the t-SNE package with learned embeddings as input. Color of a node indicates the research areas of the author. Red: “database”, blue: “data mining”, cyan: “artificial intelligence”, purple: “information retrieval”

6 Conclusions

In this paper, we have proposed a semantic-path biased representation learning framework on heterogeneous information networks. Our model demonstrates the ability of merging different semantic path (meta path). Our experiments on real-world datasets in multiple tasks show that the performance of the proposed model outperformed several state-of the art baseline algorithms. However, the procedure of searching the best weights of discriminative semantic paths needs efforts and expertise. Directions of future research include exploring the right way to tune the hyper-parameters automatically to learning better representations of the HINs; integrating deep neural networks to improve the embedding performance. One might also want to investigate the application of the SERL framework to multiple network analysis tasks.

Acknowledgments. The authors would like to thank the anonymous reviewers for their helpful comments. This work was supported by the National Natural Science Foundation of China(Grant No. 61472403, 61303243, 61702470).



References

1. Belkin, M., Niyogi, P.: Laplacian eigenmaps and spectral techniques for embedding and clustering. In: Advances in Neural Information Processing Systems, pp. 585–591 (2002)
2. Cao, S., Lu, W., Xu, Q.: GraRep: learning graph representations with global structural information. In: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, pp. 891–900. ACM (2015)
3. Chang, S., Han, W., Tang, J., Qi, G.J., Aggarwal, C.C., Huang, T.S.: Heterogeneous network embedding via deep architectures. In: Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 119–128. ACM (2015)

4. Chen, T., Sun, Y.: Task-guided and path-augmented heterogeneous network embedding for author identification. In: Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, pp. 295–304. ACM (2017)
5. Dong, Y., Chawla, N.V., Swami, A.: metapath2vec: scalable representation learning for heterogeneous networks. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 135–144. ACM (2017)
6. Fu, T.y., Lee, W.C., Lei, Z.: HIN2Vec: explore meta-paths in heterogeneous information networks for representation learning. In: Proceedings ACM on Conference on Information and Knowledge Management, pp. 1797–1806. ACM (2017)
7. Grover, A., Leskovec, J.: node2vec: scalable feature learning for networks. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 855–864. ACM (2016)
8. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint [arXiv:1301.3781](https://arxiv.org/abs/1301.3781) (2013)
9. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems, pp. 3111–3119 (2013)
10. Perozzi, B., Al-Rfou, R., Skiena, S.: DeepWalk: online learning of social representations. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 701–710. ACM (2014)
11. Shi, C., Hu, B., Zhao, W.X., Yu, P.S.: Heterogeneous information network embedding for recommendation. arXiv preprint [arXiv:1711.10730](https://arxiv.org/abs/1711.10730) (2017)
12. Sun, Y., Han, J.: Mining heterogeneous information networks: principles and methodologies. Synth. Lect. Data Mining Knowl. Discov. **3**(2), 1–159 (2012)
13. Sun, Y., Han, J., Yan, X., Yu, P.S., Wu, T.: PathSim: meta path-based top-k similarity search in heterogeneous information networks. Proc. VLDB Endowment **4**(11), 992–1003 (2011)
14. Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., Mei, Q.: Line: large-scale information network embedding. In: Proceedings of the 24th International Conference on World Wide Web, pp. 1067–1077. International World Wide Web Conferences Steering Committee (2015)
15. Tang, J., Zhang, J., Yao, L., Li, J., Zhang, L., Su, Z.: ArnetMiner: extraction and mining of academic social networks. In: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 990–998. ACM (2008)
16. Wang, C., Song, Y., Li, H., Zhang, M., Han, J.: KnowSim: a document similarity measure on structured heterogeneous information networks. In: 2015 IEEE International Conference on Data Mining (ICDM), pp. 1015–1020. IEEE (2015)
17. Wang, D., Cui, P., Zhu, W.: Structural deep network embedding. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1225–1234. ACM (2016)
18. Wang, H., Zhang, F., Hou, M., Xie, X., Guo, M., Liu, Q.: Shine: Signed heterogeneous information network embedding for sentiment link prediction. In: Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, pp. 592–600. ACM (2018)



Social Bayesian Personal Ranking for Missing Data in Implicit Feedback Recommendation

Yijia Zhang^{1,2} , Wanli Zuo^{1,2}, Zhenkun Shi^{1,2} , Lin Yue³,
and Shining Liang^{1,2}

¹ College of Computer Science and Technology, Jilin University,
Jilin 130000, China

shizkl4@mails.jlu.edu.cn

² Key Laboratory of Symbol Computation and Knowledge Engineering
of Ministry of Education, Changchun 130012, China

³ School of Computer Science and Information Technology,
Northeast Normal University, Changchun 130117, China

Abstract. Recommendation systems estimate user's preference to suggest items that might be interesting for them. Recently, implicit feedback recommendation has been steadily receiving more attention because it can be collected on a larger scale with a much lower cost than explicit feedback. The typical methods for recommendation are not well-designed for implicit feedback recommendation. Some effective methods have been proposed to improve implicit feedback recommendation, but most of them suffer from the problems of data sparsity and usually ignore the missing data in implicit feedback. Recent studies illustrate that social information can help resolve these issues. Towards this end, we propose a joint factorization model under the BPR framework utilizing social information. Remarkable, the experimental results show that our method performs much better than the state-of-the-art approaches and is capable of solving implicit problems, which indicates the importance of incorporating social information in the recommendation process to address the poor prediction accuracy.

Keywords: Implicit feedback recommendation · BPR · Social information

1 Introduction

Recommendation system plays a vital role in daily lives. With the explosion of data, people are faced with an increasingly severe "Information Overload". Recommendation Systems are information filtering systems mitigating the information overload problem by filtering vital information according to user's preferences, interests, and observed behavior about items [1]. Recommender systems help to capture users' individualized preferences using a variety of information gathering techniques [2]. User information such as reviews, ratings, and relevant feedback provided by individuals on their initiative which directly reflect user's preference is called explicit feedback. While the information that can't directly express user's preference for things such as purchase history, search mode, and click method is called implicit feedback. Implicit feedback recommendations have received considerable attention in recent years owing to implicit feedback information makes the recommendation method based on it more adaptively.

However, the implicit feedback is lack of negative feedback, where only positive feedback is available. Apart from the positive feedback, the remaining data is a mixture of real negative feedback and missing values. Therefore, it is hard to reliably infer which item a user did not like from implicit feedback, which makes it a big challenge for the recommendation. To deal with the problem of missing negative samples, several approaches have been proposed which can be roughly classified into two categories: sample-based learning and whole data-based learning. The previous samples negative feedback from the missing data, while the later treats all the missing data as negative. Therefore, sample-based approaches are more effective while whole-data based approaches provide higher coverage [3]. With the advent of online social networks, incorporating social relations into recommender systems has demonstrated potential to improve recommendation performance, and to help mitigate some public issues, such as data sparsity and cold start [4].

In this paper, we focus on implicit feedback. Moreover, we build our recommendation systems under the BPR framework utilizing social information to deal with missing data. To investigate this phenomenon, we conduct our experiment based on two well-known publicly available datasets (FilmTrust and Last.fm).

Our contributions are summarized as follows:

1. We present a novel model incorporating social relations information. In our model, we show that user relations can be considered as a specialization of implicit feedback issues and we construct the extended matrix to deal with missing data issues for implicit feedback recommendation.
2. We build our model by factorizing the interactions of user-item, user-extended item, and user-user jointly; we utilize social information as auxiliary knowledge to learn personalized ranking effectively.
3. We evaluate the proposed method on two real-world datasets, and empirical results show that the proposed model can improve recommendation performance compared to state-of-the-art methods.

The rest of this paper is organized as follows: Some related work is discussed in Sect. 2. The problem definition is presented in Sect. 3. We introduce our proposed model in Sect. 4. Our experiments are reported in Sect. 5. Finally, we conclude the paper and present some directions for future work.

2 Related Work

Social recommender systems have been widely studied, considered that a social recommender system improves the accuracy of the traditional recommendation system by taking social relations as additional inputs [5]. Koren et al. [6] proposed a model SVD++ which latent factor models and neighborhood models are merged smoothly. Ma et al. [7] design the SoRec approach by fusing the user-item rating matrix with user-user trust matrix. However, this model suffers from the problem of low interpretability. To model trust information more realistically, they further proposed RSTE, which interprets user's rating decision as the balance between user's taste and her trusted neighbors' favors [8]. Jamali et al. [9] proposed a random walk method (TrustWalker) which combines trust-based and item-based recommendation. Wang et al. [10] proposed a contextual social

network model that takes into account both participants' personal characteristics and mutual relations. Yang et al. [11] proposed a hybrid method TrustMF that combines both a truster model and a trustee model from the perspectives of trusters and trustees, both the users who trust the active user and those who are trusted by the user will influence the user's ratings on unknown items. Yang et al. [12] proposed model FIP (Friendship-Interest Propaga). In the model, a probability model is established for the relationship between user-item and user-user respectively. The author assumes that the relationship between the user and the item is depended on the distribution of visual and potential features simultaneously. Pan et al. [13] proposed a new and improved assumption called group Bayesian personalized ranking (GBPR) and designed an efficient algorithm correspondingly. Ester et al. [14] proposed model to approximate tie strength and extended the popular Bayesian Personalized Ranking (BPR) model to incorporate the distinction between strong and weak ties.

It should be noted that Zhao et al. [15] proposed a model SBPR to improve personalized ranking for collaborative filtering, our work differs from it in three ways. Firstly, our model constructs an extended matrix to deal with the missing data for implicit feedback utilizing social information, instead of paying attention to the ranking of single user's preference. Second, except for the user-item interaction, we consider user-user interaction simultaneously. Third, we build a joint model to learn the personalized ranking more effectively.

3 Preliminaries

In this section, we first introduce the implicit feedback recommendation, then formalize Bayesian-based ranking (BPR), which is designed for optimizing users' preferences over pair-wise samples.

3.1 Implicit Feedback

Let $U = [u_1, \dots, u_M] \in \mathbb{R}^{D \times M}$ denotes the user latent vectors and $V = [v_1, \dots, v_N] \in \mathbb{R}^{D \times N}$ denotes the item latent vectors, where D is the latent feature dimension, M is the number of users, N is the number of items. We define user-item interaction matrix $Y \in \mathbb{R}_{M \times N}$ as,

$$y_{ui} = \begin{cases} 1, & \text{if interaction (user } u, \text{ item } i) \text{ is observed;} \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

Here a value of 1 for y_{ui} indicates that there is an interaction between user u and item i ; however, it does not mean u actually likes i . Similarly, a value of 0 does not necessarily mean u does not like i , it can be that the user is not aware of the item.

3.2 Bayesian Personalized Ranking (BPR)

Bayesian personalized ranking (BPR) model is widely known as the state-of-the-art method to tackle the recommendation with implicit feedback [16]. The main idea of it

is to learn a personalized pairwise ranking function $p(>_u|\Theta)$ which generates a partial order [17]. The optimization objective for BPR is based on the maximum posterior estimator, and the ranking function is represented as,

$$\begin{aligned} p(v_i >_u v_j|\Theta) &= \sigma(r_{uij}) \\ &= \sigma(u^T v_i - u^T v_j) \end{aligned} \quad (2)$$

Where $\sigma = 1/(1 + e^{-x})$ is the logistic sigmoid function; Θ denotes all parameters (K -dimensional latent factors of users and items). $v_i >_u v_j$ indicates user prefers item i than item j . r is the estimate preference. In order to estimate the parameters, we minimize the following negative log-likelihood function as,

$$L_{bpr} = - \sum_{(u,v_i,v_j \in D)} \ln p(v_i >_u v_j|\Theta) + \lambda \|\Theta\|_F^2 \quad (3)$$

Where the subset D consists of training triples, λ is regularization parameter.

4 Our Proposed Approaches

In this section, we present our model for recommending with social relations information. Our first assumption is constructing the extended matrix utilizing social information. We then consider that user relations can be expressed as special implicit feedback issues. Lastly, we propose a novel joint model to learn user and item latent features effectively.

4.1 Social Information for Missing Data

Social information has been proved to have a good effect on implicit feedback issues. Due to stable and long-lasting social bindings, people tend to trust recommendations from their friends more than those from strangers [18]. Therefore, it is realistic to fill the missing data accounts for the preferences of user's friends.

In this paper, we first construct the extended user-item interaction matrix. We assume that if user's friends have interacted with the particular item that is not observed by the user, the user may prefer the item on a significant probability. Figure 1 illustrates how we construct the extended matrix using an original user-item matrix with social information to fill the missing data. As for missing data, we consider it as a weak positive instance if user's friends have observed it. We can see the extended matrix is less sparse than the original matrix and it can solve the cold start problems. The extended matrix $M \in \mathbb{R}_{M \times N}$ is define

$$m_{ui} = \begin{cases} 1, & \text{if interaction (user } u, \text{ item } i) \text{ is observed directly or indirectly;} \\ 0, & \text{otherwise.} \end{cases} \quad (4)$$

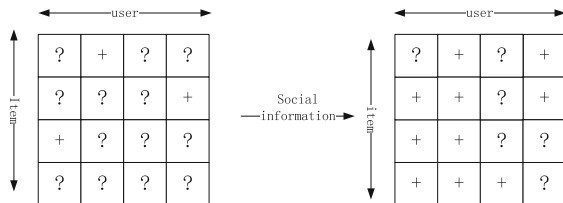


Fig. 1. The process for constructing extended matrix. On the left, the origin user-item matrix is shown, it is extended with social relation to the right matrix.

4.2 Implicit User Interactions

User-user interaction acts as an important role in implicit feedback recommendation systems. The interactions between users reflect the trust between users, and users with interactive relationships have greater similarity than others. We think user show preference for their friends, which is consistent with the user’s preference for items that he or she interacts with. We define user-user interaction matrix $S \in \mathbb{R}_{M \times M}$ as,

$$S_{ui} = \begin{cases} 1, & \text{if interaction (user } u, \text{ item } i) \text{ is observed directly or indirectly;} \\ 0, & \text{otherwise.} \end{cases} \quad (5)$$

Where a value of 1 for s_{uv} shows user u and v are known to each other, and they are friends. A value of 0 shows user u and user v have no interaction in the social network [19]. We represent user-user interaction as an inherent feedback problem by constructing the user-user interaction matrix. Moreover, our goal is to estimate the preference that user have for other users.

4.3 Joint Factorization with BPR

So far, we have developed two instantiations of our model. Figure 3 shows the main idea of the model. To make them together, we build our model to fuse the two instantiations under the BPR framework, so that they can mutually reinforce each other to learn the user latent features and item latent features. The objective function is devised as,

$$L = -\alpha \sum_{(u,i,j) \in D_y} \ln \sigma(\hat{y}_{uij}) - (1 - \alpha) \sum_{(u,i,j) \in D_m} \ln \sigma(\hat{m}_{uij}) - (1 - \alpha) \sum_{(u,i,j) \in D_s} \ln \sigma(\hat{s}_{uij}) + \lambda (\|U\|^2 + \|V\|^2) \quad (6)$$

Where D_y , D_m and D_s are the training sets for the user-item entries in the matrix $Y \in \mathbb{R}_{M \times N}$, $M \in \mathbb{R}_{M \times N}$ and user-user entry in the matrix $S \in \mathbb{R}_{M \times M}$, λ is the regularization parameter, U and V are the matrices of user and item latent features, α is the parameter to balance the performance of the three parts of the function.

It is obvious that the objective function learns a personalized ranking for recommendation jointly, and our function aims to optimize the latent features with relations as (7), where \widehat{r}_{ui} , \widehat{r}_{uj} and \widehat{r}_{uk} indicates the estimate scores of the user to the positive item, weak positive item and negative item.

$$\begin{cases} \widehat{r}_{ui} > \widehat{r}_{uj} \\ \widehat{r}_{ui} > \widehat{r}_{uk} \end{cases} \begin{cases} \widehat{r}_{ui} > \widehat{r}_{uk} \\ \widehat{r}_{uj} > \widehat{r}_{uk} \end{cases} \quad (7)$$

In sum, the first term accounts for typical user-item interaction, the second item is based on extended matrix, and the third term pays attention to user-user interaction. Since we are dealing with a ranking problem, it makes sense to use a loss function that is optimized for ranking. It has been proved that BPR is suitable for the task of ranking in social networks because it is tailored to data where only positive feedback is available [20]. And [21, 22] provided empirical evidence that factorizing the relations jointly is at least as good as the sequential approach (Fig. 2).

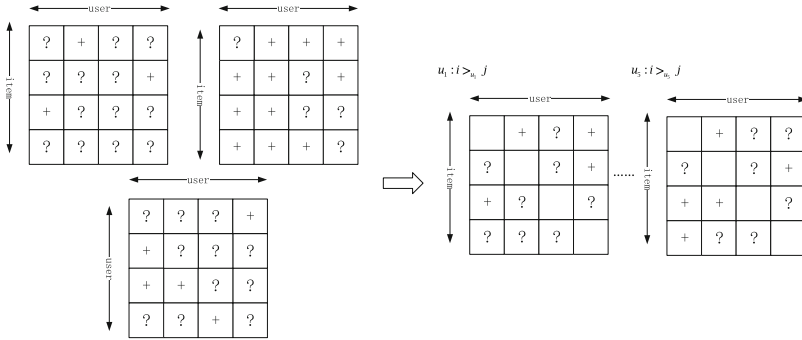


Fig. 2. The architecture of S-BPR. On the left side, there are three kinds of interactions which are user-item, user-extended item, user-user, our approach creates user-specific pairwise preferences $i >_u j$ between a pair of items. On the right side, plus (+) indicates that a user prefers item i over the item j ; minus (-) indicates that he prefers j over i

4.4 Solutions

In our method, we inherit the SGD strategy to realize our designed framework. Specifically, the optimization procedure is conducted with respect to D_y , D_m and D_s . A training instance is randomly sampled at each iteration, and a gradient descent step for all related parameters regarding the loss of the training instance is performed. Algorithm 1 details the procedure of optimization. The derivative of the loss function presented in Eq. (6) is as:

$$\frac{\partial L_{S-BPR}(\hat{y}_{uij})}{\partial \Theta} = \alpha \cdot \frac{-e^{-\hat{y}_{uij}}}{1 + e^{-\hat{y}_{uij}}} \cdot \frac{\partial \hat{y}_{uij}}{\partial \Theta} - \lambda_{\Theta} \cdot \Theta \quad (8)$$

$$\frac{\partial L_{S-BPR}(\hat{m}_{uij})}{\partial \Theta} = (1 - \alpha) \cdot \frac{-e^{-\hat{m}_{uij}}}{1 + e^{-\hat{m}_{uij}}} \cdot \frac{\partial \hat{m}_{uij}}{\partial \Theta} - \lambda_{\Theta} \cdot \Theta \quad (9)$$

$$\frac{\partial L_{S-BPR}(\hat{s}_{uij})}{\partial \Theta} = (1 - \alpha) \cdot \frac{-e^{-\hat{s}_{uij}}}{1 + e^{-\hat{s}_{uij}}} \cdot \frac{\partial \hat{s}_{uij}}{\partial \Theta} - \lambda_{\Theta} \cdot \Theta \quad (10)$$

The partial derivatives are:

$$\frac{\partial \hat{y}_{uij}}{\partial \Theta} = \begin{cases} v_{if} - v_{jf} & \text{if } \theta = u_f \\ u_f & \text{if } \theta = v_{if} \\ -u_f & \text{if } \theta = v_{jf} \\ 0 & \text{else} \end{cases} \quad (11)$$

$$\frac{\partial \hat{m}_{uij}}{\partial \Theta} = \begin{cases} v_{if} - v_{jf} & \text{if } \theta = u_f \\ u_f & \text{if } \theta = v_{if} \\ -u_f & \text{if } \theta = v_{jf} \\ 0 & \text{else} \end{cases} \quad (12)$$

$$\frac{\partial \hat{s}_{uij}}{\partial \Theta} = \begin{cases} u_{if} - u_{jf} & \text{if } \theta = u_f \\ u_f & \text{if } \theta = u_{if} \\ -u_f & \text{if } \theta = u_{jf} \\ 0 & \text{else} \end{cases} \quad (13)$$

Where f denotes the f_{th} latent features of the entry instance.

Algorithm 1 The optimization for S-BPR

- 1 Random initialize Θ ;
 - 2 Repeat
 - 3 Repeat
 - 4 Draw (u, v_i, v_j) from D_y :
 - 5 $\Theta \leftarrow \Theta + \mu \left(\alpha \cdot \frac{e^{-\hat{y}_{uij}}}{1 + e^{-\hat{y}_{uij}}} \cdot \frac{\partial \hat{y}_{uij}}{\partial \Theta} + \lambda_{\Theta} \Theta \right)$;
 - 6 Until convergence
 - 7 Repeat
 - 6 Draw (u, v_i, v_j) from D_m :
 - 7 $\Theta \leftarrow \Theta + \mu \left((1 - \alpha) \cdot \frac{e^{-\hat{m}_{uij}}}{1 + e^{-\hat{m}_{uij}}} \cdot \frac{\partial \hat{m}_{uij}}{\partial \Theta} + \lambda_{\Theta} \Theta \right)$;
 - 8 Until convergence
 - 9 Repeat
 - 10 Draw (u, u_i, u_j) from D_s :
 - 11 $\Theta \leftarrow \Theta + \mu \left((1 - \alpha) \cdot \frac{e^{-\hat{s}_{uij}}}{1 + e^{-\hat{s}_{uij}}} \cdot \frac{\partial \hat{s}_{uij}}{\partial \Theta} + \lambda_{\Theta} \Theta \right)$;
 - 12 Until convergence
 - 13 Until convergence or max-iteration has been reached;
-

5 Experiments

In this section, we conduct experiments on the two real-world datasets to demonstrate the effectiveness of the proposed method. We provide analysis of the experimental results. We also do some extensive experiments to compare the performance with different settings.

5.1 Datasets

We use two social network datasets to evaluate our models: FilmTrust and Last.fm. They are publicly accessible on the websites and used widely in the evaluation of previous trust-aware recommender systems. The statistics of four datasets are given in Table 1.

FilmTrust. This is a dataset crawled from the entire FilmTrust website in June 2011.

Last.fm.¹ This dataset contains social networking, tagging, and music artist listening information from a set of 2K users from Last.fm online music system.

Table 1. Statistics of the four statistics

Statistics	FilmTrust	Last.fm
# of users	1508	1892
# of items	2071	17632
# of ratings	35497	92834
Density	1.14%	0.27%
# of trusters	609	1892
# of trustees	732	1892
# of trusts	1853	25434
Density	0.42%	0.71%

5.2 Baselines

BPR. This is a sampling-based algorithm that optimizes the pair-wise ranking between observed instances and sampled negative instances.

MF. This a traditional method for recommendation.

STE. This a matrix factorization approach for the social network-based recommendation. Their method is a linear combination of basic matrix factorization approach and a social network-based approach.

¹ <http://ir.ii.uam.es/hetrec2011>.

MR-BPR. This method combines multi-relational matrix factorization models and BPR models based on the users’ feedback on items and social relations simultaneously.

SBPR. This method improves personalized ranking for collaborative filtering using social connections based on BPR.

5.3 Performance Comparison and Analysis

In this paper, we choose three popular metrics for implicit feedback recommendation to evaluate the performance of different models: Precision@K, NDCG@K and AUC. For all the datasets, we randomly choose 80% of each user’s ratings for training, leaving 20% of the dataset left for testing.

The optimal experimental settings for each method are determined either by our experiments or suggested by previous works. For our model, we randomly initialized model parameters with a Gaussian distribution (with a mean of 0 and standard deviation of 0.01), and we use stochastic gradient (SGD) to optimize the model. The latent feature dimension in our experiment is set as 4, the learning rate of 0.1, the regularization parameter of 0.01, the balance parameter is 0.5, the number of iteration is 30, and in order to increase the speed of optimizing, we use batch technologies where the batch size is set as 256, we conduct 256 instances at each time, the epoch number is calculated by the sum of the instances and batch size. We conduct top-10 recommendation on the FilmTrust and Last.fm dataset. As for each test instance, we choose 100 negative items randomly as negative samples.

The experimental results for top-10 recommendation are summarized in Table 2. From the result, we can see that: (1) Among the baseline methods our model performs best on both of datasets which is as expected because we use social information to fill the missing data so that we can learn the personalized ranking more effectively. (2) By utilizing social information, MR-BPR and SBPR perform better than BPR, STE performs better than MF, which shows the importance of social information for the recommendation. (3) The accuracy improvements on the two datasets are significant, especially in terms of AUC, our method performs much better than other methods. Although in terms of other metrics, our model improves less significant than SBPR, it still outperforms all the baselines in the top-10 recommendation. Thus, we can say it is effective for a recommendation in most cases.

Table 2. Performance comparison

		BPR	MF	STE	MR-BPR	SBPR	Our model
FilmTrust	AUC	0.8295	0.8220	0.8229	0.8251	0.8356	0.8467
	NDCG@10	0.3245	0.2786	0.3684	0.4204	0.4902	0.5184
	Precision@10	0.1548	0.1288	0.2682	0.2606	0.3492	0.3897
Last.fm	AUC	0.8778	0.8136	0.8090	0.8285	0.8792	0.8832
	NDCG@10	0.0291	0.0017	0.0227	0.0368	0.0387	0.0558
	Precision@10	0.0220	0.0030	0.0144	0.0334	0.0431	0.0455

To investigate the performance of our model on the different values of N , we compare these methods on three metrics where $N = [10, 20, 50, 100]$, Figs. 3, 4, 5, 6, 7 and 8 illustrates the results by varying N values on the two datasets. In is easy to notice that the recommendation accuracy on the FilmTrust dataset is decreasing as N get lager, whereas it is increasing on the Last.fm dataset. Apparently, the impact of N on the Last.fm dataset is more significant than that on the FilmTrust dataset. Figures 3 and 4 illustrates our model performs better among all baselines in terms of AUC, which means our model has a good effect on personalized ranking, and it shows the values of AUC are stable with the increase of N on both of the datasets. We also observe that BPR, MR-BPR and SBPR perform better than MF and STE, this is possibly due to the fact that the BPR framework can improve the personalized ranking of recommendation. From Figs. 5, 6, 7 and 8 we can see that the performance of our model is less significant than SBPR when the value of N is 10 and 20, but still better than it and other baselines, and our model performs much better than all the baselines with the rise of N values.

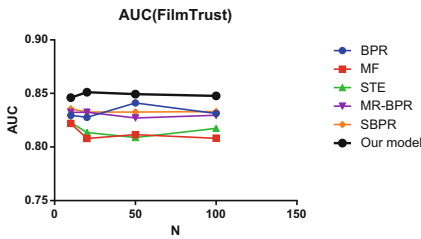


Fig. 3. AUC (FilmTrust)

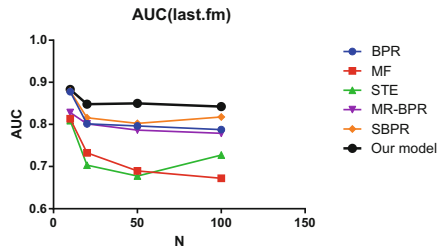


Fig. 4. AUC (last.fm)

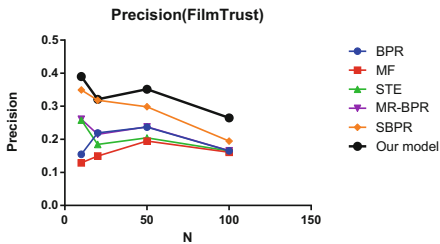


Fig. 5. Precision (FilmTrust)

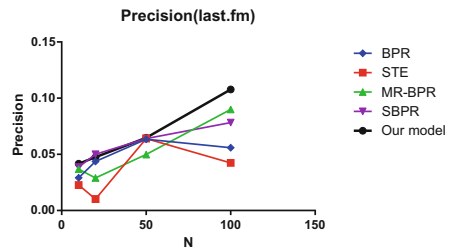


Fig. 6. Precision (last.fm)

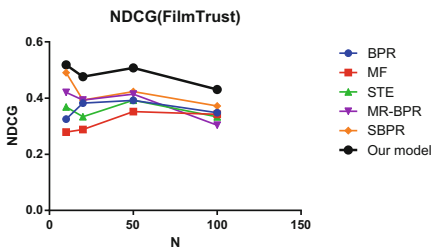


Fig. 7. NDCG (FilmTrust)

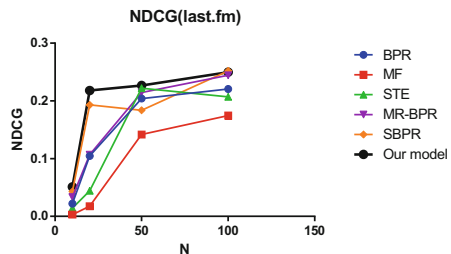


Fig. 8. NDCG (last.fm)

6 Conclusion and Future Work

In this paper, we propose a novel joint factorization model incorporating with social information. We aim at missing data issues in implicit feedback utilizing social information. Moreover, we consider user-user interaction as implicit feedback issue so that we can learn interactions between users under the framework of BPR. The experimental results show the proposed model performs better on the two real-world datasets comparing with other recommendation methods, which indicates the importance of social information for the implicit feedback recommendation. In future, we aim to find a more effective method to fill the missing data utilizing social information. We will focus on indirect relations between users considering context data of users rather than their ratings only to improve recommendation accuracy further.

Acknowledgement. This work is sponsored by the Nature Science Foundation of Jilin Province (No. 20180101330JC), the National Nature Science Foundation of China (No. 60973040, No. 61602057), the Outstanding Young Talent Project of Jilin Providence (No. 2017052005954), the Fundamental Research Funds for the Central Universities (No. 2412017QD028), China Postdoctoral Science Foundation (No. 2017M621192), the Scientific and Technological Development Program of Jilin Province (No. 20180520022JH).






References

1. Isinkaye, F.O., Folajimi, Y.O., Ojokoh, B.A.: Recommendation systems: principles, methods and evaluation. *Egypt. Inform. J.* **16**(3), 261–273 (2015)
2. Davoudi, A., Chatterjee, M.: Modeling trust for rating prediction in recommender systems. In: *SIAM Workshop on Machine Learning Methods for Recommender Systems*, pp. 1–8. SIAM (2016)
3. Chen, J., Zhang, H., He, X., et al.: Attentive collaborative filtering: multimedia recommendation with item-and component-level attention. In: *Proceedings of 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 335–344. ACM (2017)
4. Yang, B., Lei, Y., Liu, D., Liu, J.: Social collaborative filtering by trust. In: *Proceedings of IJCAI International Joint Conference on Artificial Intelligence*, pp. 2747–2753 (2013)
5. Liu, F., Lee, H.J.: Use of social network information to enhance collaborative filtering performance. *Expert Syst. Appl.* **37**(7), 4772–4778 (2010)
6. Koren, Y.: Factorization meets the neighborhood: a multifaceted collaborative filtering model. In: *Proceedings of 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 426–434. ACM (2008)
7. Ma, H., Yang, H., Lyu, M.R., King, I.: SoRec: social recommendation using probabilistic matrix factorization. In: *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, pp. 0–9 (2008)
8. Ma, H., King, I., Lyu, M.R.: Learning to recommend with social trust ensemble. In: *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval SIGIR*, pp. 203–210 (2009)
9. Jamali, M., Ester, M.: TrustWalker: a random walk model for combining trust-based and item-based recommendation. In: *Proceedings of 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 397–406. ACM (2009)

10. Wang, Y., Li, L., Liu, G.: Social context-aware trust inference for trust enhancement in social network based recommendations on service providers. *World Wide Web* **18**(1), 159–184 (2015)
11. Yang, B.; Lei, Y.; Liu, D., Liu, J.: Social collaborative filtering by trust. In: Proceedings of 23rd International Joint Conference on Artificial Intelligence (IJCAI), pp. 2747–2753. AAAI Press (2013)
12. Yang, S.H., Long, B., Smola, A., et al.: Like like alike: joint friendship and interest propagation in social networks. In: Proceedings of International Conference on World Wide Web, pp. 537–546 (2011)
13. Pan, W., Chen, L.: GBPR: group preference based Bayesian personalized ranking for one-class collaborative filtering. In: Proceedings of International Joint Conference on Artificial Intelligence, pp. 2691–2697 (2013)
14. Wang, X., Lu, W., Ester, M., et al.: Social recommendation with strong and weak ties. In: Proceedings of 25th ACM International on Conference on Information and Knowledge Management, pp. 5–14. ACM (2016)
15. Zhao, T., McAuley, J., King, I.: Leveraging social connections to improve personalized ranking for collaborative filtering. In: Proceedings of 23rd ACM International Conference on Conference on Information and Knowledge Management, pp. 261–270. ACM (2014)
16. Rendle, S., Freudenthaler, C., Gantner, Z., et al.: BPR: Bayesian personalized ranking from implicit feedback. In: Proceedings of 20th Conference on Uncertainty in Artificial Intelligence, pp. 452–461. AUAI Press (2009)
17. Chen, J., Wang, C., Wang, J., et al.: Recommendation for repeat consumption from user implicit feedback. *IEEE Trans. Knowl. Data Eng.* **28**(11), 3083–3097 (2015)
18. Yang, X., Guo, Y., Liu, Y., Steck, H.: A survey of collaborative filtering based social recommender systems. *Comput. Commun.* **41**, 1–10 (2014)
19. He, X., Liao, L., Zhang, H., Nie, L., Hu, X., Chua, T.S.: Neural collaborative filtering. In: Proceedings of 26th International Conference on World Wide Web, pp. 173–182. International World Wide Web Conferences Steering Committee (2017)
20. Cao, D., Nie, L., He, X., et al.: Embedding factorization models for jointly recommending items and user generated lists. In: Proceedings of 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 585–594. ACM (2017)
21. Artus, K., Lucas, D., Christoph, F., Lars, S.: Multi-relational matrix factorization using Bayesian personalized ranking for social network data. In: Proceedings of WSDM, pp. 173–182. ACM (2012)
22. Shi, Z., Zuo, W., Chen, W., Yue, L., Han, J., Feng, L.: User relation prediction based on matrix factorization and hybrid particle swarm optimization. In: Proceedings of 26th International Conference on World Wide Web, pp. 1335–1341. ACM (2017)



A Semantic Path-Based Similarity Measure for Weighted Heterogeneous Information Networks

Chunxue Yang^(✉) , Chenfei Zhao , Hengliang Wang , Riming Qiu ,
Yuan Li , and Kedian Mu

School of Mathematical Sciences, Peking University, Beijing 100871, China
cxueyoung@gmail.com

Abstract. In recent years, recommender systems based on heterogeneous information networks (HIN) have gained wide attention. In order to generate more attractive recommendations, weighted heterogeneous information network (WHIN) has been proposed, which attaches attribute values to links. The widely-used similarity measures for HIN may fail to capture the semantics of weighted meta-path. This makes designing a similarity measure specially for WHIN more necessary. In this paper, we propose a semantic path-based similarity measure called Wgt-Sim, which is a generalization of PathSim presented by Sun et al. Furthermore, to demonstrate the capability of WgtSim in capturing semantics, we apply WgtSim to recommender system on WHIN to predict ratings given by users. The experiments on two real datasets show that the recommender system with WgtSim outperforms that with previous measures.

Keywords: Heterogeneous information network · Similarity measure
Recommender system

1 Introduction

Recently, many data mining tasks have been exploited on heterogeneous information network (HIN), and recommendation is not an exception. Distinguished from homogeneous information network, HIN is a directed graph involving multi-typed objects and multi-typed links denoting different relations [9]. Organizing a recommender system as a HIN, we can easily obtain the complex relations among users, items, groups and so on. Figure 1(a) shows the schema of HIN on Yelp¹ data.

As is known to all, what users think of the items has a significant impact on recommendations. Unfortunately, we can not learn it from HIN. The rating score on an item given by a user represents how he or she likes the item. To integrate

¹ Yelp is a website which publishes crowd-sourced reviews about local businesses.
<https://www.yelp.com/>.

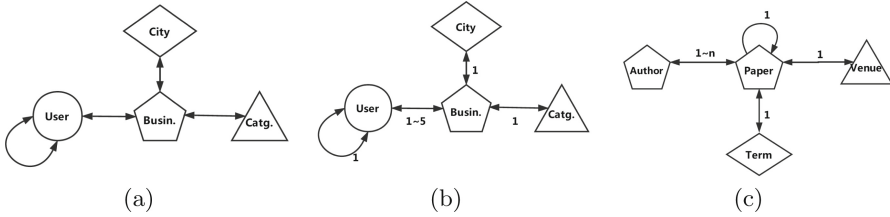


Fig. 1. Network schemas of HIN and WHIN.

this kind of information into HIN, Shi et al. proposed weighted heterogeneous information network (WHIN) and weighted meta-path [7]. Links in WHIN are attached to attribute values. The network schema of WHIN constituted by Yelp data is shown in Fig. 1(b). It contains two types of attribute values. One of them is attached to the rating relation between users and businesses, the links of which take values from 1 to 5, and the other type is on all the relations except rating relation, the links of which all take value 1.

With the advent of WHIN, it becomes essential to study how to measure the similarity between objects in this new kind of network, since similarity measure is a fundamental function in many applications including recommendation. Intuitively, similar users are likely to be attracted by the same item. We usually recommend items to a user according to his or her similar users' taste.

Most existing similarity measures are defined for HIN. If we directly apply them to WHIN, attribute values will be ignored. This may lead to poor performance in recommendation. To illustrate this, consider an extreme example. Both Alice and Bob have seen a certain film, and they will be taken for similar users by similarity measures proposed on HIN, such as PathSim [10], PCRW [2] and HeteSim [4], because of the same viewing record. Actually, Alice loves it and rates 5, but it's out of Bob's favor. Bob rates 1. They have totally different tastes. To solve this kind of problem, Shi et al. [7] designed a computation strategy based on previous measures, where the same viewing record will be ignored if the scores given by the two users are different. But the strategy can cause loss of some important information during computation and lead to data sparsity.

To the best of our knowledge, there is no similarity measure proposed specially for WHIN to make use of this structure's advantages. In this paper, we introduce a concept of degree of contribution to similarity, which is an attribute of path instance. Based on this concept, we propose a semantic path-based similarity measure for WHIN called WgtSim, which is a generalization of PathSim. In recommendation tasks, WgtSim of two users is figured out by comparing their preferences on items. The major contributions of this paper are summarized as follows:

- We propose the notion of degree of contribution to similarity. This attribute of symmetric path instance refers to how important this path instance is when we measure the similarity of the two objects at ends of it.
- We propose a semantic path-based similarity measure for WHIN called WgtSim. WgtSim makes good use of attribute values in WHIN, which enables it measure similarity of objects belong to the same type accurately.

- Empirical studies in recommendation tasks on two real datasets, Yelp and MovieLens, are conducted to validate the effectiveness of WgtSim. The results show that recommender systems can achieve better performances with WgtSim.

2 Related Work

HIN [9], which involves multi-typed objects and multi-typed links denoting different relations, provides a paradigm to manage networked data. A sequence of relations between two object types in HIN is defined as a meta-path [10]. One of the most important characteristics of HIN is the rich semantics of meta-paths. For example, the Business-City-Business path in Fig. 1(a) means businesses in the same city.

As an effective semantic capturing tool [6], meta-path has been widely used in many data mining tasks in HIN, such as similarity measure [4, 5, 10]. Sun et al. [10] propose a path-based similarity measure called PathSim. It's defined among the same type of objects, and able to find peer objects in the network (e.g. find authors in the similar field and with similar reputation), which turns out to be meaningful in many scenarios.

Recently, the importance of HIN for recommendations has been increasingly recognized by researchers [8, 11, 13]. However, conventional HIN lose some of information. For example, the rating score on an item given by a user, to which is exactly attached importance in a recommender system, is not contained in HIN. Shi et al. [7] propose WHIN and weighted meta-path to integrate this kind of information.

A weighted meta-path is denoted as $A_1 \xrightarrow{\delta(R_1)} A_2 \xrightarrow{\delta(R_2)} \dots \xrightarrow{\delta(R_l)} A_{l+1}$, where A_i represents an object type, R_j represents a relation and $\delta(R_j)$ is the range of attribute values on relation R_j , ($i = 1, 2, \dots, l + 1; j = 1, 2, \dots, l$). We say a concrete path $p = a_1 \xrightarrow{w_1} a_2 \xrightarrow{w_2} \dots \xrightarrow{w_l} a_{l+1}$ is a path instance of the weighted meta-path if each link $e_i = \langle a_i, a_{i+1} \rangle \in R_i$, each object $a_i \in A_i$, and each attribute value $w_i \in \delta(R_i)$.

The similarity measures for HIN may fail to capture semantics of weighted meta-path. Shi et al. [7] design a similarity computation strategy to make the existing path-based similarity measures still usable in WHIN, which only takes the path instances that satisfy the given constraint into consideration. But there is some information lying in the path instances that do not satisfy the given constraint. This motivate us to propose a novel similarity measure for WHIN to make more use of information from weighted meta-paths.

3 Semantic Path-Based Similarity Measures

3.1 PathSim: A Path-Based Similarity Measure for HIN

PathSim is a meta-path-based similarity measure for HIN, which was proposed by Sun et al. [10].

Since in many scenarios, finding similar objects in networks is to find similar peers, PathSim is confined on the symmetric meta-paths. A meta-path $P = A_1A_2\dots A_l$ is symmetric if P is equal to P^{-1} , such as User-Business-User and User-Business-Category-Business-User.

Given a symmetric meta-path P , PathSim between two objects of the same type x and y , denoted as $s(x, y)$, is given as follows:

$$s(x, y) = \frac{2 \times |\{p_{x \rightsquigarrow y} : p_{x \rightsquigarrow y} \in P\}|}{|\{p_{x \rightsquigarrow x} : p_{x \rightsquigarrow x} \in P\}| + |\{p_{y \rightsquigarrow y} : p_{y \rightsquigarrow y} \in P\}|} \quad (1)$$

where $p_{x \rightsquigarrow y}$ is a path instance between objects x and y , $p_{x \rightsquigarrow x}$ is that between x and x , and $p_{y \rightsquigarrow y}$ is that between y and y .

Equation 1 shows that given a meta-path P , PathSim of x and y is determined by their connectivity and visibility, where their connectivity is defined as the number of path instances between them following P , and the visibility is defined as the number of path instances between themselves.

3.2 A Similarity Computation Strategy on Weighted Meta-paths

In order to adapt previous HIN-based similarity measures to WHIN, Shi et al. [7] designed a computation strategy.

They add certain constraints on attribute values of weighted meta-path. The constrained meta-paths used to calculate degree of similarity of users in their experiments on Yelp data are like this:

$$\text{User} \xrightarrow{i} \text{Business} \xrightarrow{j} \text{User} | i = j \quad (2)$$

In a weighted meta-path, if all attribute value ranges $\delta(R_i)$ take a specific value, it is an atomic meta-path. Every meta-path has a group of atomic meta-paths. For example, $\text{User} \xrightarrow{1} \text{Business} \xrightarrow{1} \text{User}$, ..., $\text{User} \xrightarrow{5} \text{Business} \xrightarrow{5} \text{User}$ are all atomic meta-paths of constrained meta-path Eq. 2.

The computation strategy is to count the number of path instances of each atomic meta-path that satisfy the given constraint and sum them up as path counts of the constrained meta-path $P|C$. Then constrained PathSim of object x and y , denoted as $s(x, y)$, can be calculated as:

$$s(x, y) = \frac{2 \times |\{p_{x \rightsquigarrow y} : p_{x \rightsquigarrow y} \in P|C\}|}{|\{p_{x \rightsquigarrow x} : p_{x \rightsquigarrow x} \in P|C\}| + |\{p_{y \rightsquigarrow y} : p_{y \rightsquigarrow y} \in P|C\}|}$$

PathSim is a special case of the similarity computation strategy on weighted meta-path proposed by Shi et al., when the constraint on relations is \emptyset .

Essentially, this strategy means that only part of atomic meta-paths count, precisely, those satisfy given constraint C . And other atomic meta-paths are ignored. In the task of recommendation on Yelp, only the path instances where two users have exactly the same ratings on a certain business contribute to the similarity between these two users.

3.3 WgtSim: A Semantic Path-Based Similarity Measure for WHIN

Although the similarity measure strategy proposed by Shi et al. makes an attempt to adapt PathSim to WHIN, there are still something can be improved. First, many atomic meta-paths should not be ignored, despite they dissatisfy the given constraint. For example, User $\xrightarrow{5}$ Business $\xrightarrow{4}$ User is a non-negligible clue when we measure similarity of users on Yelp. Second, different atomic meta-paths make different degrees of contribution when we measure the similarity between the objects at the both ends. If we hope to recommend attractive businesses to users instead of picking out those they don't care about, atomic meta-path User $\xrightarrow{5}$ Business $\xrightarrow{5}$ User should be attached more importance than User $\xrightarrow{1}$ Business $\xrightarrow{1}$ User. We hope that high similarity results from users liking the same businesses, rather than disliking the same, so that we can recommend businesses to a user according to his similar users' taste. These motivated us to propose a novel meta-path-based similarity measure for WHIN.

Without loss of generality, we assume the attribute values on links are finite and discrete.

Definition 1 (Relative difference ratio). *In a path instance $p = a_1 \xrightarrow{w_1} a_2 \xrightarrow{w_2} \dots \xrightarrow{w_l} a_{l+1}$ of a symmetric meta-path $P = A_1 \xrightarrow{\delta(R_1)} A_2 \xrightarrow{\delta(R_2)} \dots \xrightarrow{\delta(R_l)} A_{l+1}$, links e_i and e_{l+1-i} belong to the same relation R_i , $i = 1, 2, \dots, l$. For $i \in \{1, \dots, l\}$, the relative difference ratio of the link pair (e_i, e_{l+1-i}) , denoted RD_i , is defined as:*

$$RD_i = \frac{D_i - |w_i - w_{l+1-i}|}{D_i}$$

where D_i is the difference between the maximum and the minimum of $\delta(R_i)$.

For path instances of meta-path $P = \text{User} \xrightarrow{1\sim 5} \text{Business} \xrightarrow{1\sim 5} \text{User}$, the smaller relative difference ratio means the more two users' opinions differ on the business.

In some cases, for example, where the values represent users' attitudes to items, the larger the value on the link is, the more the path instance contributes to similarity of users. Since when we want to recommend attractive items to users, we care much about what they like but little about their dislikes.

However, sometimes the opposite is the case. The relation between authors and papers in bibliographic networks (as shown in Fig. 1(c)) can take values (e.g., 1, 2, 3...) which represent the order of authors of the paper. The smaller the value on the link is, the more the path instance contributes to similarity of authors.

Definition 2 (Absolute difference ratio). *In a path instance $p = a_1 \xrightarrow{w_1} a_2 \xrightarrow{w_2} \dots \xrightarrow{w_l} a_{l+1}$ of a symmetric meta-path $P = A_1 \xrightarrow{\delta(R_1)} A_2 \xrightarrow{\delta(R_2)} \dots \xrightarrow{\delta(R_l)} A_{l+1}$, links e_i and e_{l+1-i} belong to the same relation R_i , $i = 1, 2, \dots, l$. For $i \in \{1, \dots, l\}$, the absolute difference ratio of the link pair (e_i, e_{l+1-i}) , denoted AD_i , is defined as:*

$$AD_i = \frac{2M_i - w_i - w_{l+1-i}}{M_i}$$

and if the latter, the absolute difference ratio is defined as

$$AD_i = \frac{w_i + w_{l+1-i} - 2m_i}{M_i}$$

where M_i is the maximum of $\delta(R_i)$, and m_i is the minimum of $\delta(R_i)$.

Definition 3 (Degree of contribution to similarity). *The degree of contribution to similarity of path instance $p = a_1 \xrightarrow{w_1} a_2 \xrightarrow{w_2} \dots \xrightarrow{w_l} a_{l+1}$, denoted as $c(p)$, is defined as:*

$$c(p) = \exp \left\{ - \sum_{i=1}^{\frac{l}{2}} \frac{AD_i^{\alpha_i}}{RD_i^{\beta_i}} \right\} \tag{3}$$

where $\alpha_i \geq 0$ is absolute coefficient and $\beta_i \geq 0$ is relative coefficient.

The larger α_i is set, the more importance is attached to absolute values on links. The larger β_i is set, the more importance is attached to relative difference between values on a pair of links.

When $\alpha_1 = \beta_1 = 1.5$, the degree of contribution of $p = user \xrightarrow{i} business \xrightarrow{j} user$ is shown in Table 1. In general, $c(p)$'s correlation with $|i + j|$ is positive, but negative with $|i - j|$. The former is due to our little care about users' dislikes in recommendation task. The latter shows that the more different two users' opinions on the business are, the less the path instance contributes to their similarity. When $i = j = 5$, $c(p)$ reaches its maximum 1. When $i = 5, j = 1$ or $i = 1, j = 5$, $c(p)$ is 0, that is to say, the two users' opinions are so different that the path instance does not contribute to similarity at all.

Table 1. Degree of contribution of $p = user \xrightarrow{i} business \xrightarrow{j} user$ ($\alpha_1 = \beta_1 = 1.5$).

$c(p) \begin{matrix} \backslash j \\ i \end{matrix}$	5	4	3	2	1
5	1	0.8714	0.4890	0.0243	0
4	0.8714	0.7765	0.4890	0.1321	0.0004
3	0.4890	0.4890	0.4889	0.2145	0.0243
2	0.0243	0.1321	0.2145	0.2686	0.0780
1	0	0.0004	0.0243	0.0780	0.1321

When $D_i = 0$, that is to say, the attribute values do not vary on the links of relation R_i , we specify value of $AD_i^{\alpha_i} / RD_i^{\beta_i}$ as 0. It means relation R_i have no effect on the degree of contribution. For example, for path instances of meta-path $P = User \xrightarrow{1 \sim 5} Business \xrightarrow{1} City \xrightarrow{1} Business \xrightarrow{1 \sim 5} User$, only the attribute values on the links of relation between User and Business determine the degree of contribution.

Definition 4 (WgtSim). Given a symmetric weighted meta-path P , $WgtSim$ between x and y of the same type, denoted as $s(x, y)$, is defined as:

$$s(x, y) = \frac{2 \times \sum_{\{p_{x \rightsquigarrow y} : p_{x \rightsquigarrow y} \in P\}} c(p_{x \rightsquigarrow y})}{\sum_{\{p_{x \rightsquigarrow x} : p_{x \rightsquigarrow x} \in P\}} c(p_{x \rightsquigarrow x}) + \sum_{\{p_{y \rightsquigarrow y} : p_{y \rightsquigarrow y} \in P\}} c(p_{y \rightsquigarrow y})}$$

Some properties of $WgtSim$, such as non-negativity, symmetry and conditional self-maximum, are shown in Theorem 1.

Theorem 1. Given a symmetric meta-path $P = P_l P_l^{-1} = A_1 \xrightarrow{\delta(R_{1l})} \dots \xrightarrow{\delta(R_{2l})} A_{2l+1}$, P_l has $K = \prod_{i=1}^l |\delta(R_i)|$ atomic meta-paths. They are Ap_1, Ap_2, \dots, Ap_K . $WgtSim$ satisfies the following properties:

1. Non-negativity: $s(x, y) \geq 0$.
2. Symmetry: $s(x, y) = s(y, x)$.
3. Conditional self-maximum: If matrix $C_{K \times K}(i; j) = c(Ap_i Ap_j^{-1})$ is positive definite, $WgtSim$ is self-maximum. $s(x, y) \leq 1$, and $s(x, x) = 1$.

Proof

- (1) Since $c(p) \geq 0$.
- (2) $s(x, y) = s(y, x)$ if and only if

$$\sum_{\{p_{x \rightsquigarrow y} : p_{x \rightsquigarrow y} \in P\}} c(p_{x \rightsquigarrow y}) = \sum_{\{p_{y \rightsquigarrow x} : p_{y \rightsquigarrow x} \in P\}} c(p_{y \rightsquigarrow x}).$$

Consider map τ

$$\begin{aligned} \tau : \{p_{x \rightsquigarrow y} : p_{x \rightsquigarrow y} \in P\} &\rightarrow \{p_{y \rightsquigarrow x} : p_{y \rightsquigarrow x} \in P\} \\ p &\mapsto p^{-1} \end{aligned}$$

where p^{-1} means the reverse path instance of p . τ is a bijection between the two sets. For any path instance $p = x \xrightarrow{w_1} a_1 \xrightarrow{w_2} \dots \xrightarrow{w_{l-1}} a_{l-1} \xrightarrow{w_l} y$ of a symmetric meta path P ,

$$c(p) = \exp \left\{ - \sum_{i=1}^{\frac{l}{2}} \frac{AD_i^{\alpha_i}}{RD_i^{\beta_i}} \right\} = \exp \left\{ - \sum_{i=\frac{l}{2}+1}^l \frac{AD_i^{\alpha_i}}{RD_i^{\beta_i}} \right\} = c(p^{-1}) = c(\tau(p))$$

Map τ keeps the same degree of contribution to similarity. Therefore,

$$\sum_{\{p_{x \rightsquigarrow y} : p_{x \rightsquigarrow y} \in P\}} c(p_{x \rightsquigarrow y}) = \sum_{\{p_{y \rightsquigarrow x} : p_{y \rightsquigarrow x} \in P\}} c(p_{y \rightsquigarrow x}).$$

- (3) For any object $o \in A_{l+1}$, there are $a_i^{(o)}$ path instances of atomic meta-path Ap_i between x and o , and $b_i^{(o)}$ path instances of Ap_i^{-1} between o and y , $i = 1, 2, \dots, K$. Then

$$s(x, y) = \frac{2 \sum_{\{o: o \in A_{l+1}\}} \sum_{i=1}^K \sum_{j=1}^K c(Ap_i Ap_j^{-1}) a_i^{(o)} b_j^{(o)}}{\sum_{\{o: o \in A_{l+1}\}} \sum_{i=1}^K \sum_{j=1}^K c(Ap_i Ap_j^{-1}) (a_i^{(o)} a_j^{(o)} + b_i^{(o)} b_j^{(o)})}$$

Let

$$f(b_1^{(o)}, b_2^{(o)}, \dots, b_K^{(o)}) = \sum_{i=1}^K \sum_{j=1}^K c(Ap_i Ap_j^{-1}) (a_i^{(o)} a_j^{(o)} + b_i^{(o)} b_j^{(o)} - 2a_i^{(o)} b_j^{(o)}) \quad (4)$$

Then

$$f'_{b_j^{(o)}}(b_1^{(o)}, b_2^{(o)}, \dots, b_K^{(o)}) = 2 \sum_{i=1}^K c(Ap_i Ap_j^{-1}) (b_i - a_i), i = 1, 2, \dots, K \quad (5)$$

$$f''_{b_i^{(o)} b_j^{(o)}} = 2c(Ap_i Ap_j^{-1}), i = 1, 2, \dots, K, j = 1, 2, \dots, K$$

Substitute $b_j = a_j, j = 1, 2, \dots, K$ into Eqs. 4 and 5, the values of them are all 0. Meanwhile, Hessian matrix is positive definite. Then we know for any $o \in A_{l+1}$,

$$\frac{2 \sum_{i=1}^K \sum_{j=1}^K c(Ap_i Ap_j^{-1}) a_i^{(o)} b_j^{(o)}}{\sum_{i=1}^K \sum_{j=1}^K c(Ap_i Ap_j^{-1}) (a_i^{(o)} a_j^{(o)} + b_i^{(o)} b_j^{(o)})} \leq 1$$

Then $s(x, y) \leq 1$, and obviously $s(x, x) = 1$.

3.4 Discussions

Both PathSim and the similarity computation strategy proposed by Shi et al. are special cases of WgtSim. If the links of all the relations on meta-path $P = A_1 \xrightarrow{\delta(R_1)} \dots \xrightarrow{\delta(R_l)} A_{l+1}$ take value 1, i.e. $\delta(R_1) = \dots = \delta(R_l) = \{1\}$, then value of $AD_i^{\alpha_i} / RD_i^{\beta_i}$ is 0 by definition, $i = 1, 2, \dots, l$. So for any path instance p of meta path P , $c(p) = 1$. WgtSim degenerates into PathSim.

Take constrained meta-path Eq. 2 in Yelp data as an example. If we let α_1 be 0 and β_1 converge to $+\infty$ in Eq. 3, the degree of contribution of path instance $p = user \xrightarrow{i} business \xrightarrow{j} user$ to similarity will be

$$c(p) = \begin{cases} e^{-1} & i = j \\ 0 & \text{otherwise} \end{cases}$$

In this situation, WgtSim becomes an equivalent of the strategy proposed by Shi et al.

4 Experiments

In this section, experiments on two real datasets are conducted to validate the effectiveness of WgtSim. We apply WgtSim to recommender system on WHIN to predict the scores that a user rates on items. The recommender system used in experiments is based on the method SemRec proposed by Shi et al. [7].

4.1 Datasets

We use two real datasets in experiments: MovieLens² and Yelp³ datasets. MovieLens dataset published by GroupLens research group contains 2113 users, 10197 movies and 855598 ratings, which has dense rating relations but sparse social relations. Yelp dataset includes 10000 users, 10167 business with 49872 ratings, and it has dense social relations but sparse rating relations. The detailed information of these two datasets is shown in Table 2.

Table 2. Statistics of MovieLens/Yelp datasets

Dataset	Relations (A – B)	Number of A	Number of B	Number of (A – B)	Ave. degrees of A/B
Movielens	User-movie	2113	10197	855598	404.9/84.6
	Movie-genre	10197	72	10197	1.0/144.3
	Movie-country	10109	20	20670	2.0/1033.5
	Movie-director	10068	4060	10068	1.0/2.5
Yelp	User-business	10000	10167	49872	4.99/4.91
	User-user	3938	3938	28684	7.28/7.28
	Business-category	10167	324	10167	1/31.38
	Business-city	10119	878	29985	2.96/34.15

4.2 Metrics

We use mean absolute error (MAE) and root-mean-square error (RMSE) to evaluate the performance of rating prediction. The smaller MAE or RMSE means the better performance.

$$MAE = \frac{\sum_{(u,i) \in R_{test}} |R_{u,i} - \hat{R}_{u,i}|}{|R_{test}|}$$

$$RMSE = \sqrt{\frac{\sum_{(u,i) \in R_{test}} (R_{u,i} - \hat{R}_{u,i})^2}{|R_{test}|}}$$

where R_{test} represents the test set, $R_{u,i}$ represents the rating of the user u on item i , and $\hat{R}_{u,i}$ represents the predicted rating of user u on item i by a certain measure.

² <https://grouplens.org/datasets/hetrec-2011/>.

³ <http://www.yelp.com/dataset/>.

4.3 Baseline Models

We compare the recommender system with WgtSim with three methods. These baseline methods are illustrated below.

- **PMF** [3]: Probabilistic Matrix Factorization (PMF) products lower-rank users matrix U and items matrix V by user-item matrix R , and predict the ratings of users on items with matrix $\hat{R} = U^T V$. PMF utilizes the user-item matrix for recommendations without considering meta-paths.

The recommendation method SemRec proposed by Shi et al. [7] can figure out similar users of a target user by similarity matrix S . The score that the target user rates on an item can be inferred through the rating scores of his similar users on the item. The similarity matrix S can be calculated by the following methods.

- **PathSim** [10]: PathSim, which has been illustrated in Sect. 3.1, is a similarity measure based on meta-path without considering attribute values on links.
- **Constrained PathSim** [7]: Constrained PathSim, whose main idea has been introduced in Sect. 3.2, adds certain constrains on attribute values to make PathSim usable on WHIN.
- **WgtSim**: A semantic path-based similarity measure for WHIN proposed by us, which is a generalization of PathSim and constrained PathSim.

4.4 Recommendation Effectiveness

For each dataset, we define four meaningful meta-paths, of which detailed description can be seen in Table 3. We make four different partitions of datasets, which are 60%, 70%, 80% and 90%. For example, the proportion is 80% means we randomly split the datasets into training and test ones, and 80% of the whole data are used for training and the remaining 20% are for testing. The process is repeated five times and the average MAE and RMSE of the five rounds are reported. The parameters are set to make all methods to achieve the best performances, specifically $\alpha_1 = 2$ and $\beta_1 = 12$ in Eq. 3.

Table 3. Meta-paths used in experiments

MovieLens	Yelp
$U \xrightarrow{1\sim 5} M \xrightarrow{1\sim 5} U$	$U \xrightarrow{1} U \xrightarrow{1} U$
$U \xrightarrow{1\sim 5} M \xrightarrow{1} \text{Genre} \xrightarrow{1} M \xrightarrow{1\sim 5} U$	$U \xrightarrow{1\sim 5} B \xrightarrow{1\sim 5} U$
$U \xrightarrow{1\sim 5} M \xrightarrow{1} \text{Country} \xrightarrow{1} M \xrightarrow{1\sim 5} U$	$U \xrightarrow{1\sim 5} B \xrightarrow{1} \text{Catg.} \xrightarrow{1} B \xrightarrow{1\sim 5} U$
$U \xrightarrow{1\sim 5} M \xrightarrow{1} \text{Director} \xrightarrow{1} M \xrightarrow{1\sim 5} U$	$U \xrightarrow{1\sim 5} B \xrightarrow{1} \text{City} \xrightarrow{1} B \xrightarrow{1\sim 5} U$

The rating prediction results can be seen in Table 4⁴. We can see that comparing to PMF, which only uses the rating matrix, SemRec with PathSim, which

⁴ Since the Yelp dataset in the CIKM paper [7] has not been published, we use another Yelp dataset in our experiments, which has sparser ratings than CIKM-Yelp (The density of rating matrix in CIKM-Yelp is reported in [12]). Thus the performance of Constrained PathSim is different from what they reported in their paper.

Table 4. Rating prediction MAE and RMSE for four methods (The improvement is based on PMF).

Dataset	Training settings	PMF		PathSim		Cons. PathSim		WgtSim	
		MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
MovieLens	60%	0.9459	1.2943	0.8358 11.6%	0.9743 24.7%	0.8260 12.7%	0.9642 33.0%	0.6861 27.5%	0.8658 33.1%
	70%	0.9437	1.2937	0.8327 11.8%	0.9715 24.9%	0.8218 12.9%	0.9603 25.8%	0.6850 27.4%	0.8643 33.2%
	80%	0.9365	1.2850	0.8290 11.9%	0.9669 24.8%	0.8172 12.7%	0.9554 25.6%	0.6842 26.9%	0.8631 32.8%
	90%	0.9281	1.2743	0.8255 11.1%	0.9632 24.4%	0.8105 12.7%	0.9477 25.6%	0.6824 26.5%	0.8606 32.5%
Yelp	60%	1.3189	1.7516	1.1389 13.6%	1.4800 15.5%	1.2453 5.6%	1.6055 8.3%	1.0389 21.2%	1.3499 22.9%
	70%	1.3140	1.7483	1.1351 13.6%	1.4793 15.4%	1.2332 6.1%	1.5918 9.0%	1.0382 21.0%	1.3442 23%
	80%	1.3092	1.7464	1.1349 13.3%	1.4766 15.4%	1.2274 6.2%	1.5826 9.4%	1.0332 21.1%	1.3407 23.2%
	90%	1.3018	1.7451	1.1348 12.8%	1.4714 15.7%	1.2146 6.7%	1.5692 10.1%	1.0258 21.2%	1.3352 23.5%

uses additional heterogeneous information by meta-paths, are significantly better. Note that the performance of PathSim is worse than constrained PathSim on MovieLens dataset, the reason is that, conventional PathSim fails to avoid noise caused by atomic meta-paths where the relative difference of attribute values is large. However, for Yelp data, the rating relation is rather sparse, and adding constraints to meta-paths exacerbates the sparsity. Therefore, PathSim outperforms constrained PathSim on Yelp data.

When comparing the results of WgtSim and constrained PathSim, the good performance of WgtSim may be attributed to two reasons. First, WgtSim takes atomic meta-paths like $\text{User} \xrightarrow{5} \text{Business} \xrightarrow{4} \text{User}$ into consideration, which alleviates the data sparsity problem. Second, WgtSim gives different weights (i.e. the contribution to similarity) to different path instances according to the attribute values on them, so it can subtly capture the information of weighted meta-paths.

In general, the experimental results show that the methods based on weighted meta-paths perform better than those based on unweighted ones. Among measures based on weighted meta-paths, WgtSim proves to be more effective in capturing semantics.

5 Conclusion

In this paper, we analyzed the inapplicability of traditional similarity measures for HIN and the limitations of the computation strategy proposed by Shi et al. on WHIN. Then we proposed a novel similarity measure called WgtSim, which

generalizes PathSim to WHIN. WgtSim emphasizes the degree of contribution to similarity in WHIN, and takes both absolute and relative difference ratios of attribute values into consideration. Experiments on two real datasets illustrate that WgtSim performs better in capturing semantics of weighted meta-path compared with baseline methods. In addition, the similarity measure we proposed can be applied on other data mining tasks such as classification on WHIN.

Acknowledgements. This work was partly supported by the National Natural Science Foundation of China under Grant No. 61572002, No. 61170300, No. 61690201, and No. 61732001.

References

1. Bu, S., Hong, X., Peng, Z., Li, Q.: Integrating meta-path selection with user-preference for top-k relevant search in heterogeneous information networks. In: Proceedings of the 18th IEEE International Conference on Computer Supported Cooperative Work in Design, pp. 301–306 (2014)
2. Lao, N., Cohen, W.W.: Fast query execution for retrieval models based on path-constrained random walks. In: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 881–888. ACM (2010)
3. Salakhutdinov, R., Mnih, A.: Probabilistic matrix factorization. In: Proceedings of the 20th International Conference on Neural Information Processing Systems, pp. 1257–1264. Curran Associates Inc. (2007)
4. Shi, C., Kong, X., Huang, Y., Yu, P.S., Wu, B.: HeteSim: a general framework for relevance measure in heterogeneous networks. *IEEE Trans. Knowl. Data Eng.* **26**(10), 2479–2492 (2014)
5. Shi, C., Kong, X., Yu, P.S., Xie, S., Wu, B.: Relevance search in heterogeneous networks. In: Proceedings of the 15th International Conference on Extending Database Technology, EDBT 2012, pp. 180–191. ACM (2012)
6. Shi, C., Yu, P.S.: *Heterogeneous Information Network Analysis and Applications*. DA. Springer, Cham (2017)
7. Shi, C., Zhang, Z., Luo, P., Yu, P.S., Yue, Y., Wu, B.: Semantic path based personalized recommendation on weighted heterogeneous information networks. In: Proceedings of the 24th ACM International Conference on Information and Knowledge Management, pp. 453–462. ACM (2015)
8. Shi, C., Zhou, C., Kong, X., Yu, P.S., Liu, G., Wang, B.: HeteRecom: a semantic-based recommendation system in heterogeneous networks. In: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1552–1555. ACM (2012)
9. Sun, Y., Han, J.: Mining heterogeneous information networks: a structural analysis approach. *SIGKDD Explor. Newsl.* **14**(2), 20–28 (2013)
10. Sun, Y., Han, J., Yan, X., Yu, P.S., Wu, T.: Pathsim: meta path-based top-k similarity search in heterogeneous information networks. *Proc. VLDB Endow.* **4**, 992–1003 (2011)
11. Yu, X., et al.: Recommendation in heterogeneous information networks with implicit user feedback. In: Proceedings of the 7th ACM Conference on Recommender Systems, pp. 347–350. ACM (2013)

12. Zhao, H., Yao, Q., Li, J., Song, Y., Lee, D.L.: Meta-graph based recommendation fusion over heterogeneous information networks. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2017, pp. 635–644. ACM (2017)
13. Zheng, J., Liu, J., Shi, C., Zhuang, F., Li, J., Wu, B.: Recommendation in heterogeneous information network via dual similarity regularization. *Int. J. Data Sci. Anal.* **3**, 35–48 (2017)



Cross-Domain Recommendation for Mapping Sentiment Review Pattern

Yang Xu, Zhaohui Peng, Yupeng Hu, Xiaoguang Hong^(✉),
and Wenjing Fu

School of Computer Science and Technology, Shandong University,
Jinan, People's Republic of China
{xuyang0211, pzh, huyupeng, hxg}@sdu.edu.cn,
fuwenjing1214@mail.sdu.edu.cn

Abstract. Cross-domain algorithms which aim to transfer knowledge available in the source domains to the target domain are gradually becoming more attractive as an effective approach to help improving quality of recommendations and to alleviate the problems of cold-start and data sparsity in recommendation systems. However, existing works on cross-domain algorithm mostly consider ratings, tags and the text information like reviews, cannot use the sentiments implicated in the reviews efficiently. In this paper, we propose a Sentiment Review Pattern Mapping framework for cross-domain recommendation, called SRPM. The proposed SRPM framework can model the semantic orientation of the reviews of users, and transfer sentiment review pattern of users by using a multi-layer perceptron to capture the nonlinear mapping function across domains. We evaluate and compare our framework on a set of Amazon datasets. Extensive experiments on each cross-domain recommendation scenarios are conducted to prove the high accuracy of our proposed SRPM framework.

Keywords: Cross-domain recommendation · Sentiment review pattern
Pattern mapping

1 Introduction

Cross-domain recommendation systems are gradually becoming more attractive as a practical approach to improve quality of recommendations and to alleviate cold-start problem, especially in small and sparse datasets. These algorithms mine knowledge on users and items in a source domain to improve the quality of the recommendations in a target domain. They can also provide joint recommendations for items belonging to different domains by the linking information among these domains [1]. Most existing works about cross domain recommendation tend to aggregate knowledge from different domains from the perspective of explicitly specified common information [2–4] or transferring latent features [5, 8, 9, 13]. However, the aggregated knowledge merely based on ratings, tags, or the text information like reviews, ignores the sentiments implicated in the reviews. After watching a popular film, using a novel electronic product or playing a video game, users often rate them and submit reviews to share their feelings, which could convey fairly rich sentiment information.

For all we know, existing cross-domain recommendation algorithms which utilize user reviews didn't take full advantage of the sentiment information of these reviews. They implement knowledge transfer by mixing positive and negative reviews together, which will weaken and even lose some sentiment information of the users, especially the negative sentiment. For instance, a user may deeply care about the plot of a novel, and he made positive comments on the plots of some novels in the domain of electronic book (source domain) while made negative comments on the plots of some other novels. If we transfer the knowledge gained from user reviews from the source domain to the target domain without distinguishing the sentiment polarity of these reviews, some latent factors such as "plot", "positive sentiment" and "negative sentiment" of the reviews will be mixed up as "users' feature" to be transferred to the target domain. In the domain of movie (target domain), a movie with poor plots, namely the movie whose latent factors "plot" and "negative sentiment" take higher weight, will produce a match with the users' feature transferred to the target domain. Nevertheless, the user may not be fond of this movie.

To address this problem, in this paper, we propose a new cross-domain recommendation framework called SRPM. Under SRPM, we can effectively identify the sentiment orientation of user reviews and adapt topic modeling approach to deduce the sentiment review pattern (SRP) from user reviews. To achieve the goal of transferring knowledge, we propose an MLP based mapping method to transfer sentiment information of users from source domain to target domain. Then we can get an affine SRP for a cold-start user in the target domain and predict the cold-start user's rating of items in the target domain. Through transferring SRP of users, the SRPM method gets a superior performance in cross-domain recommendation.

To summarize, the major contributions of this paper are as follows:

- We consider sentiment information in the cross-domain recommendation task and propose a novel cross-domain recommendation framework named SRPM. SRPM can be used to transfer sentiment review pattern from source domain to target domain and make recommendation for cold-start users in target domain.
- In SRPM, we design a sentiment information extracting approach, and propose the modeling and mapping method of sentiment review pattern.
- We systematically compare the proposed SRPM approach with other algorithms on the Amazon dataset. The results confirm that our new method substantially improves the performance of cross-domain recommendation.

SRPM is applicable for User-Item overlap scenarios in which users or items are found to be in common in both domains. In this paper, we introduce SRPM under the user overlap scenario.

The rest of this paper is organized as follows. Section 2 presents some notations and the problem formulation. Section 3 introduce the modeling method of SRP and Sect. 4 details the mapping method of SRP and the cross-domain recommendation approach. Experiments and discussion are given in Sect. 5. Section 6 reviews the related works on cross-domain recommendation and sentiment analysis in recommendation system. Conclusions are drawn in Sect. 7.

2 Preliminaries

In this section, we first introduce some notations in SRPM cross-domain recommendation framework and then present the SRPM framework to solve the cold-start recommendation problem.

2.1 Notations

Objects to be recommended in the cross-domain recommendation system are referred to as *items*. Let $\mathcal{U} = \{u_1, u_2, \dots, u_{|\mathcal{U}|}\}$ denote the set of common users in both domains and $\mathcal{J}_S = \{i_1, i_2, \dots, i_{|\mathcal{J}_S|}\}$, $\mathcal{J}_T = \{l_1, l_2, \dots, l_{|\mathcal{J}_T|}\}$ are the sets of items (e.g. movies, books, or electronics) in source domain and in target domain respectively. The user review dataset is represented as $SR_U = \{r_{u_1}, r_{u_2}, \dots, r_{u_{|\mathcal{U}|}}\}$ in source domain and TR_U in target domain, where r_{u_i} is all of reviews of user u_i in the corresponding domain. Similarly, we let $TR_I = \{r_{i_1}, r_{i_2}, \dots, r_{i_{|\mathcal{J}_T|}}\}$ denote the item review dataset in target domain, where r_{i_j} is all of the reviews which item i_j acquired in target domain.

In the SRPM framework, the sentiment analysis algorithm is employed on the review datasets mentioned above to divide them into corresponding positive review datasets (e.g. SR_U^{pos}, TR_U^{pos} and TR_I^{pos}) and negative review datasets (e.g. SR_U^{neg}, TR_U^{neg} and TR_I^{neg}). $S_U^{pos} = \{\theta_{S,u_1}^{pos}, \theta_{S,u_2}^{pos}, \dots, \theta_{S,u_{|\mathcal{U}|}}^{pos}\}$ represents the positive review pattern matrix in latent space of all users in source domain, where θ_{S,u_i}^{pos} is the positive review pattern of user u_i in source domain, similarly to S_U^{neg}, T_U^{pos} and T_U^{neg} . In addition, $T_I^{pos} = \{\theta_{T,i_1}^{pos}, \theta_{T,i_2}^{pos}, \dots, \theta_{T,i_{|\mathcal{J}_T|}}^{pos}\}$ denotes the positive review topic distribution matrix in latent space of all items in target domain, where θ_{T,i_j}^{pos} is the positive review topic distribution of item i_j in target domain, similarly to T_I^{neg} . For a cold-start user u' in target domain, $\theta_{S,u'}^{pos}$ ($\theta_{S,u'}^{neg}$) denotes the positive (negative) review pattern of user u' in source domain, and $\hat{\theta}_{T,u'}^{pos}$ ($\hat{\theta}_{T,u'}^{neg}$) represents the affine positive (negative) review pattern of user u' in target domain.

2.2 Problem Formulation

Given two domains which share the same users U . Users appearing in only one domain can be regarded as the cold-start users U' in the other domain. Without loss of generality, one domain is referred to as the source domain and the other as the target domain. The most common cross-domain recommendation approaches focus on transferring information based on ratings, tags and reviews from source domain to target domain, without accounting for any emotional information implicated the reviews.

We are tackling cross-domain recommendation task for cold-start users by modeling the Sentiment Review Pattern (SRP) of users and transferring them from source domain to target domain.

To achieve this purpose, we propose a cross-domain recommendation framework called SRPM. This framework contains three major steps, i.e., sentiment review pattern modeling, sentiment review pattern mapping and cross-domain recommendation, as illustrated in Fig. 1.

In the first step, we apply SO-CAL [7] to analyze the Semantic Orientation (SO) of each sentence of user reviews in both domains. Then, the original review datasets of both domains are divided into corresponding positive review datasets and negative review datasets respectively. Next, we employ Smoothed Latent Dirichlet Allocation (SLDA) on the sentiment tagged datasets to find the sentiment review pattern of users. In the second step, we model the cross-domain relationships of users through a mapping function based on Multi-Layer Perceptron (MLP) [6]. We assume that there is an underlying mapping relationship between the user’s SRPs of the source and target domains, and further use a mapping function to capture this relationship. Finally, in the third step, we make recommendation for cold-start user in the target domain. We can get an affine SRP for cold-start user in the target domain, with the SRP learned for him/her in the source domain and the MLP-based mapping functions between the source and target domain. In the rest of this paper, we will introduce each step of SRPM in details.

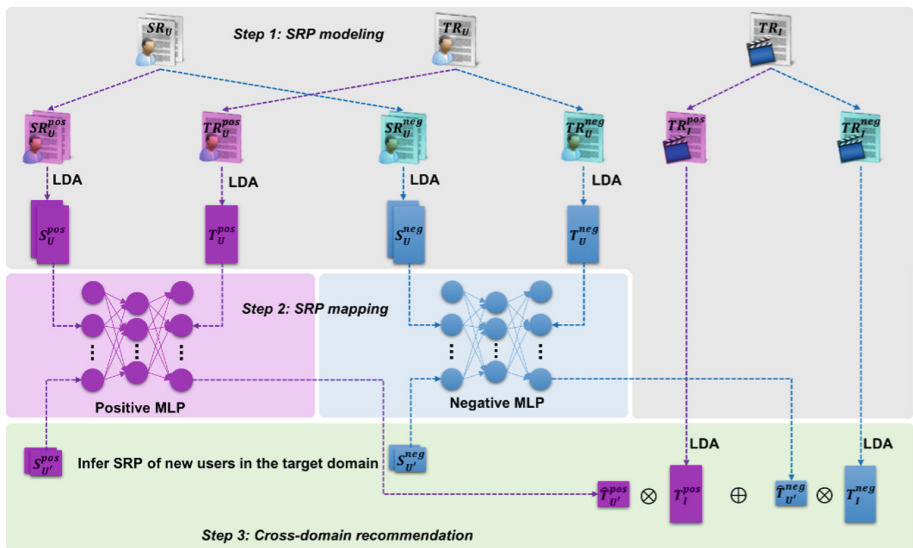


Fig. 1. Overview of the SRPM cross-domain recommendation framework

3 Sentiment Review Pattern Modeling

As discussion in the previous section, in order to transfer sentiment review pattern in the source domains to the target domain, the first phase of SRPM is to model the sentiment review pattern of common users in both domains. The key challenge is how to extract the user’s focus on the item and the positive or negative emotions expressing

in the user reviews. To address this challenge, we propose a sentiment review pattern modeling method based on sentence-level sentiment analysis approach and smoothed LDA.

3.1 Sentiment Analysis

The sentiment analysis problem in SRPM can be formulated as follows: Given a set of reviews R , a sentiment classification algorithm classifies each sentence of a piece of review $r \in R$ into one of the two classes, positive R^{pos} and negative R^{neg} . For this purpose, we apply the sentiment analysis algorithm SO-CAL [7] to analyze the semantic orientation of each sentence of user reviews. Since the sentiment analysis algorithm is not the focus of this paper, we refer the readers to the related literature such as [7, 20] for details.

We employ SO-CAL on original review sets SR_U, TR_U, TR_I to divide them into positive review subsets $SR_U^{pos}, TR_U^{pos}, TR_I^{pos}$ and negative review subsets $SR_U^{neg}, TR_U^{neg}, TR_I^{neg}$ on the sentence-level respectively.

3.2 Sentiment Review Pattern

Sentiment review pattern indicates the user’s focus on the item and the positive or negative emotions expressed in a sentence. In this paper, we use the smoothed LDA topic model [10] to extract review pattern. In our formulation, “document” is a collection of positive (negative) reviews of a user u or an item i in a certain domain, which represented as $r_u^{pos}(r_u^{neg})$ for the user u and $r_i^{pos}(r_i^{neg})$ for the item i . In the higher level, “corpus” is a collection of positive (negative) “documents” of the user set U or the item set I in that domain.

Here, we employ SLDA on the positive (negative) corpus of U in source domain $SR_U^{pos}(SR_U^{neg})$ to find the positive (negative) topic distribution $\theta_{S_u}^{pos}(\theta_{S_u}^{neg})$ of each user u in source domain and the per-topic word distribution $\beta_{S_{U,k}}^{pos}(\beta_{S_{U,k}}^{neg})$.

Similarly, we employ SLDA on $TR_U^{pos}, TR_U^{neg}, TR_I^{pos}, TR_I^{neg}$ to find topic distributions $\theta_{T_u}^{pos}, \theta_{T_u}^{neg}, \theta_{T_i}^{pos}, \theta_{T_i}^{neg}$ and per-topic word distributions $\beta_{T_{U,k}}^{pos}, \beta_{T_{U,k}}^{neg}, \beta_{T_{I,k}}^{pos}, \beta_{T_{I,k}}^{neg}$ respectively. In this paper, the topic distribution of user u ’s positive (negative) reviews is called as user u ’s Positive Review Pattern PRP_u (or Negative Review Pattern NRP_u). Then, the sentiment review pattern of user u is denoted as $SRP_u = (PRP_u, NRP_u)$.

4 Sentiment Review Pattern Mapping and Cross-Domain Recommendation

4.1 Sentiment Review Pattern Mapping

In this paper, we utilize an MLP-based method to tackle the SRP mapping problem, as shown in Fig. 1. To avoid mutual interference between positive and negative emotion factors during the process of knowledge transfer, two MLP models, the positive MLP model and the negative MLP model, were employed to map PRP and NRP from source

domain to target domain respectively. Next, we will introduce the proposed mapping algorithm under *PRP* mapping scenario, and the mapping algorithm under *NRP* mapping scenario is similar.

In our proposed mapping algorithm, only the common users with sufficient review data are used to learn the mapping function in order to guarantee its robustness to noise caused by review data sparsity and imbalance in both domains. We use entropy and statistical method to measure the cross-domain degree of common users. Formally, the cross-domain degree is defined as follows:

$$c(u) = (-p_{u,s} \log_2 p_{u,s} - p_{u,t} \log_2 p_{u,t}) \frac{\mathcal{N}(u,s) + \mathcal{N}(u,t)}{\sum_{u_i \in U_c} \mathcal{N}(u_i,s) + \mathcal{N}(u_i,t)} \quad (1)$$

where $p_{u,s} = \frac{\mathcal{N}(u,s)}{\mathcal{N}(u,s) + \mathcal{N}(u,t)}$, $p_{u,t} = \frac{\mathcal{N}(u,t)}{\mathcal{N}(u,s) + \mathcal{N}(u,t)}$

$\mathcal{N}(u, s)$ is the number of reviews in source domain of common user u , and $\mathcal{N}(u, t)$ is that in target domain. U_c denotes the set of common users between both domains. The common users with $c(u) > threshold \gamma$ are selected to learn the mapping function.

Let $\theta^S = \{\theta_1^S, \theta_2^S, \dots, \theta_N^S\}$ denotes the set of *PRP* s in the source domain, and $\theta^T = \{\theta_1^T, \theta_2^T, \dots, \theta_N^T\}$ represents the set of *PRP* s in the target domain. N is the number of common users in both domains. Under the MLP model setting, we formulate the *PRP* mapping problem as: Given N training instance (θ_i^S, θ_i^T) , $\theta_i^S, \theta_i^T \in R^M$, ($i = 1, 2, \dots, N$), where $\theta_i^S = (\theta_{i1}^S, \theta_{i2}^S, \dots, \theta_{iM}^S)$ is the *PRP* of common user u_i in the source domain and $\theta_i^T = (\theta_{i1}^T, \theta_{i2}^T, \dots, \theta_{iM}^T)$ is that in the target domain, our task is to learn an MLP mapping function to map the *PRP* from the source domain to the target domain.

In a feedforward MLP model, the output o_{ik} is formulated as

$$y_{ik} = \sum_{j=1}^L c_{jk} a_j, \quad o_{ik} = g(y_{ik}) \quad (2)$$

where c_{jk} represents the weight of the j 'th input of the output layer neuron k and L is the number of hidden neurons in each hidden layer. $g(y)$ is the activation function of the output layer, which is set to be the softmax function in this study. a_j denotes the j 'th hidden neuron activation of lower hidden layer, which can be defined as

$$y_j = \sum_{p=1}^P w_{pj} a_p, \quad a_j = f(y_j) \quad (3)$$

where w_{pj} is the weight of the p 'th input of the hidden layer neuron j (the hidden bias can be included in the input weights) and a_p is the input θ_{ip}^S or the p 'th hidden neuron activation of the lower hidden layer. P represents the number of inputs or neurons in the lower layer. $f(y)$ is the hidden layer activation function, which is set to be the ReLU function in this study.

Considering that input and output of MLP model in this study are all topic distributions, the error between θ_i^T and $o_i = (o_{i1}, o_{i2}, \dots, o_{iM})$ is measured by KL divergence:

$$E = \sum_{i=1}^N \sum_{k=1}^M o_{ik} \log \frac{o_{ik}}{\theta_{ik}^T} \quad (4)$$

To obtain the MLP mapping function, we utilize stochastic gradient descent to learn the weights. We refresh the weights of the MLP by looping through the training instances. The back-propagation algorithm is adopted to calculate the gradients of the weights, thus we can get the positive MLP mapping function $f_{pmlp}(\cdot; \theta_p)$, where θ_p is its weights set. Similarly, we can learn the negative MLP mapping function $f_{nmlp}(\cdot; \theta_n)$ by employing the above learning algorithm.

4.2 Cross-Domain Recommendation

Given a cold-start user in the target domain, we do not have sufficient information to estimate his/her preference features to make recommendation directly in the target domain. However, we can get the affine SRP for him/her in the target domain, with the SRP learned in the source domain and the MLP mapping functions from the source domain to the target domain. In this section, we will introduce how to predict the cold-start user's ratings on the specific items in the target domain.

Given a cold-start user u' in the target domain, we can extract user u' 's positive review pattern $\theta_{S,u'}^{pos}$ and negative review pattern $\theta_{S,u'}^{neg}$ from S_U^{pos} and S_U^{neg} in the source domain respectively. Then the affine positive review pattern $\hat{\theta}_{T,u'}^{pos}$ and the affine negative review pattern $\hat{\theta}_{T,u'}^{neg}$ can be obtained by the following equations:

$$\hat{\theta}_{T,u'}^{pos} = f_{pmlp}(\theta_{S,u'}^{pos}; \theta_p), \quad \hat{\theta}_{T,u'}^{neg} = f_{nmlp}(\theta_{S,u'}^{neg}; \theta_n) \quad (5)$$

Next, the similarity of each pair of topics between the corresponding emotional review dataset are defined as:

$$SIM^{pos} = \{sim_{i,j}^{pos}\}, \text{ where } sim_{i,j}^{pos} = \cos(\beta_{T_{u,i}}^{pos}, \beta_{T_{i,j}}^{pos}), \quad i, j = 1, 2, \dots, M \quad (6)$$

$$SIM^{neg} = \{sim_{i,j}^{neg}\}, \text{ where } sim_{i,j}^{neg} = \cos(\beta_{T_{u,i}}^{neg}, \beta_{T_{i,j}}^{neg}), \quad i, j = 1, 2, \dots, M \quad (7)$$

Then, the predicted emotional rating between cold-start user u' and item $l_j \in \mathcal{J}_T$ in the target domain is calculated as:

$$E(u') = \{e_{u',j}\} = \hat{\theta}_{T,u'}^{pos} \cdot SIM^{pos} \cdot T_I^{posT} - \hat{\theta}_{T,u'}^{neg} \cdot SIM^{neg} \cdot T_I^{negT} \quad (8)$$

Finally, we combine a baseline estimate function and the predicted emotional rating to predict the overall rating between u' and t_j , which is formulated as:

$$S(u', t_j) = b_{\mathcal{I}_T} + b_{u'} + b_{t_j} + e_{u',j} \quad (9)$$

where $b_{\mathcal{I}_T}$ denoted the overall average ratings of all items in the target domain. The parameter $b_{u'}$ is the user rating bias in the source domain and b_{t_j} is the item rating bias in the target domain, which indicate the observed deviations of user u' and item t_j from the average.

5 Experiments

We have conducted a set of experiments to examine the performance of our cross-domain recommendation method compared with the baselines. In this section, we first introduce the experimental settings, and then analyze the evaluation results.

5.1 Experimental Settings

Data Description. We employ Amazon cross-domain dataset [11] in our experiment. This dataset contains product reviews and star ratings with 5-star scale from Amazon, including 142.8 million reviews spanning May 1996 – July 2014. We select the top three domains with the most widely used in previous studies to employ in our cross-domain experiment. In our experiments, source domains are selected by calculating their relevance to the target domain. The relevance between two domains is defined as the ratio of the number of overlapped users' reviews in the target domain to the number of reviews in the target domain. The global statistics of these domains used in our experiments are shown in Table 1.

Table 1. Characteristics of datasets

Dataset	Books	Electronics	Movies & TV
# of Users	603,668	192,403	123,960
# of Items	367,982	63,001	50,052
# of Reviews	8,898,041	1,689,188	1,697,533
Density	0.004%	0.014%	0.027%

Experiment Setup. The domains in the Amazon dataset only have user overlaps. Thus, we evaluate the validity and efficiency of SRPM on the cross-domain recommendation task under the user overlap scenario. We randomly remove all the rating information of a certain proportion of users in the target domain and take them as cross-domain cold-start users for making recommendation. For the sake of stringency of the experiments, we set different proportions of cold-start users, namely, $\phi = 20\%$, 50% and 70% . Moreover, we repeatedly sample users for 10 times to generate different sets to balance the effect of different sets of cold-start users on the final

recommendation results. We report the average results and standard deviations over these 10 different sets. Dimension of latent factor used in the compared method and the number of topics in LDA are set as $M = 20, 50$ and 100 . For the mapping function, we set the structure of MLP as three hidden layers with $2M$ nodes in each hidden layer.

Compared Methods. We examine the performance of the proposed SRPM framework by comparing it with the following baseline methods:

- MF: This is the single-domain matrix factorization algorithm proposed in [12]. Comparing MF with the cross-domain methods will show us whether adding the extra information from source domains will increase recommendation accuracy.
- AVG: It predicts ratings by the following equation: $r_{ui} = b_T + b_u + b_i$ where b_T is the overall average ratings of all items in the target domain, b_u denotes the user rating bias in the source domain and b_i represents item bias in the target domain.
- CMF: This is a cross-domain recommendation method proposed in [14]. In CMF, the latent factors of users are shared between source domain and target domain.
- MF-MLP: This is a cross-domain recommendation framework based on MF and MLP, which is proposed by [15]. In our experiments, for MF-MLP, the structure of the MLP is set as one-hidden layer, and the number of nodes in the hidden layer is set as $2M$.

Evaluation Metric. We adopt the metrics of Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) to evaluate our method. They are defined as:

$$RMSE = \sqrt{\sum_{r_{ui} \in \mathcal{O}_{test}} \frac{(\hat{r}_{ui} - r_{ui})^2}{|\mathcal{O}_{test}|}}, \quad MAE = \frac{1}{|\mathcal{O}_{test}|} \sum_{r_{ui} \in \mathcal{O}_{test}} |\hat{r}_{ui} - r_{ui}| \quad (10)$$

where \mathcal{O}_{test} is the set of test ratings, r_{ui} denotes an observed rating in the test set, and \hat{r}_{ui} represents the predictive value of r_{ui} . $|\mathcal{O}_{test}|$ is the number of test ratings.

Table 2. Recommendation performance in terms of MAE on the “Books-Movies&TV”

	ϕ	MF	CMF	MF_MLP	AVE	SRPM
K = 20	20%	0.8783	0.8472	0.8421	0.8371	0.8016
	50%	0.8809	0.8620	0.8619	0.8810	0.8442
	70%	0.8824	0.8649	0.8638	0.9243	0.8671
K = 50	20%	0.9047	0.8902	0.8896	0.8371	0.8013
	50%	0.9372	0.9092	0.9008	0.8810	0.8388
	70%	0.9548	0.9329	0.9017	0.9243	0.8512
K = 1000	20%	1.0241	0.9088	0.9008	0.8371	0.8024
	50%	1.0764	0.9267	0.9322	0.8810	0.8429
	70%	1.1453	0.9702	0.9631	0.9243	0.8698

Table 3. Recommendation performance in terms of RMSE on the “Books-Movies&TV”

	ϕ	MF	CMF	MF_MLP	AVE	SRPM
K = 20	20%	1.3417	1.2035	1.1739	1.2280	1.1554
	50%	1.3443	1.2356	1.2090	1.2595	1.1890
	70%	1.3481	1.2360	1.2094	1.2835	1.2027
K = 50	20%	1.3698	1.2712	1.2371	1.2280	1.1223
	50%	1.4032	1.3488	1.2530	1.2595	1.1645
	70%	1.4256	1.3912	1.2544	1.2835	1.1982
K = 1000	20%	1.5069	1.4709	1.2791	1.2280	1.1776
	50%	1.5634	1.5106	1.2850	1.2595	1.1964
	70%	1.6310	1.5561	1.3282	1.2835	1.2035

Table 4. Recommendation performance in terms of MAE on the “Electronics-Movies&TV”

	ϕ	MF	CMF	MF_MLP	AVE	SRPM
K = 20	20%	0.9152	0.7091	0.8746	0.8778	0.8494
	50%	0.9175	0.7429	0.9297	0.9369	0.8729
	70%	0.9234	0.7633	0.9416	0.9461	0.9040
K = 50	20%	1.0651	0.8092	0.9699	0.8778	0.8534
	50%	1.1375	0.8495	0.9765	0.9369	0.8922
	70%	1.1958	0.9030	0.9544	0.9461	0.9270
K = 100	20%	1.2751	0.9022	1.0632	0.8778	0.8698
	50%	1.3622	0.9462	1.0316	0.9369	0.8938
	70%	1.4387	0.9883	1.0048	0.9461	0.9285

Table 5. Recommendation performance in terms of RMSE on the “Electronics-Movies&TV”

	ϕ	MF	CMF	MF_MLP	AVE	SRPM
K = 20	20%	1.4663	1.3083	1.3994	1.2403	1.1908
	50%	1.4703	1.3334	1.4447	1.2641	1.2145
	70%	1.4794	1.3509	1.4572	1.2861	1.2434
K = 50	20%	1.6300	1.3332	1.4996	1.2403	1.1948
	50%	1.7071	1.3778	1.5021	1.2641	1.2246
	70%	1.7682	1.4047	1.5113	1.2861	1.2591
K = 100	20%	1.7494	1.3422	1.5338	1.2403	1.1991
	50%	1.8300	1.3893	1.5726	1.2641	1.2248
	70%	1.8950	1.4347	1.5929	1.2861	1.2612

5.2 Performance Comparison

Recommendation Performance. Experimental results of MAE and RMSE on the two pair of domains “Books-Movies&TV” and “Electronics-Movies&TV” are presented in Tables 2, 3, 4 and 5, respectively. The domain “Books” and “Electronics” are chosen as the target domain in each pair of domains because they are extremely sparse.

We respectively evaluate all the methods under different K and ϕ in both pair of domains. From these tables, we can see that the proposed SRPM outperforms all baseline models in terms of both MAE and RMSE metrics. With the proportion of cold-start users becoming higher, the performance of single domain method MF will become progressively worse while the cross-domain methods keep satisfactory results, which shows the effectiveness of knowledge transfer. Compared with CMF and MF_MLP, our method SRPM gets an improvement of 5% to 10% both in RMSE and MAE. These results demonstrate that the SRPM is more suitable for making recommendations to cold-start users compared to other cross-domain baseline methods, especially in the dataset with high sparsity. SRPM performs better than AVG especially in higher ϕ , which demonstrates that the SRP transferred from the source domain is highly effective. And MF_MLP outperforms MF, indicating that the MLP based mapping function is feasible in knowledge transfer. For the proposed SRPM method, the optimal value of K is nearly 50 in “Books-Movie&TV” and nearly 20 in “Electronics-Movies&TV”.

6 Related Work

Existing works about cross-domain algorithm mostly extract domain-specific information from ratings [5, 12], tags [2] and the text information like reviews [16]. Ren [8] proposed the PCLF model to learn the shared common rating pattern across multiple rating matrices and the domain-specific rating patterns from each domain. Fang [2] exploited the rating patterns across multiple domains by transferring the tag co-occurrence matrix information. Xin [16] exploited review text by learning a non-linear mapping on users’ preferences on different topics across domains. On the whole, the main difference between our work and the previous approaches is the utilization of sentiment analysis method and mapping function which can predict the sentiment review pattern of cold-start users in the target domain and make cross-domain recommendations.

Sentiment analysis is widely used in recommendation systems. Computing the sentiment orientation of a user review has been studied by several researchers. Diao [17] built a language model component in their proposed JMARS model to capture aspects and sentiments hidden in reviews. Zhang [18] extracted explicit product features and user opinions by phrase-level sentiment analysis on user reviews to generate explainable recommendation results. Li [19] proposed a SUIT model to simultaneously utilize the textual topic and user item factors for sentiment analysis. In this paper, we employ sentiment analysis on cross-domain recommendation task and focus on discovering user’s sentiment review pattern and mapping it from the source domain to the target domain.

7 Conclusions

The user reviews contain plenty of sentiment information. We proposed a novel framework for cross domain recommendation that establishes linkages between the source and target domains by using sentiment review pattern of users. In this paper,

a sentiment review pattern extracting algorithm was proposed. We employed smoothed LDA and MLP based mapping method to model user's SRP and map it to the target domain to make recommendations for cold-start users. In different scenarios, that is to say, experiments convincingly demonstrate that the proposed SRPM framework can significantly improve the quality of cross-domain recommendation and SRPs extracted from reviews are important links between each domain.

Acknowledgments. This work is supported by NSF of Shandong, China (Nos. ZR2017MF065, ZR2018MF014), the Science and Technology Development Plan Project of Shandong, China (No. 2016GGX101034).




References

1. Cremonesi, P., Tripodi, A., Turrin, R.: Cross-domain recommender systems. In: ICDM 2012, pp. 496–503 (2012)
2. Fang, Z., Gao, S., Li, B., Li, J.: Cross-domain recommendation via tag matrix transfer. In: ICDM 2015, pp. 1235–1240 (2015)
3. Chen, W., Hsu, W., Lee, M.L.: Making recommendations from multiple domains. In: KDD 2013, pp. 892–900 (2013)
4. Yang, D., He, J., Qin, H., Xiao, Y., Wang, W.: A graph-based recommendation across heterogeneous domains. In: CIKM 2015, pp. 463–472 (2015)
5. Li, B., Yang, Q., Xue, X.: Transfer learning for collaborative filtering via a rating-matrix generative model. In: ICML 2009, pp. 617–624 (2009)
6. Ruck, D.W., Rogers, S.K., Kabrisky, M., Oxley, M.E., Suter, B.W.: The multilayer perceptron as an approximation to a bayes optimal discriminant function. *IEEE Trans. Neural Netw.* **1**(4), 296–298 (1990)
7. Taboada, M., Brooke, J., Tofiloski, M., Voll, K., Stede, M.: Lexicon-based methods for sentiment analysis. *Comput. Linguist.* **37**(2), 267–307 (2011)
8. Ren, S., Gao, S., Liao, J., Guo, J.: Improving cross-domain recommendation through probabilistic cluster-level latent factor model. In: AAAI 2015, pp. 4200–4201 (2015)
9. Gao, S., Luo, H., Chen, D., Li, S., Gallinari, P., Guo, J.: Cross-domain recommendation via cluster-level latent factor model. In: Blockeel, H., Kersting, K., Nijssen, S., Železný, F. (eds.) *ECML PKDD 2013. LNCS (LNAI)*, vol. 8189, pp. 161–176. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-40991-2_11
10. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. *J. Mach. Learn. Res.* **3**, 993–1022 (2003)
11. He, R., McAuley, J.: Ups and downs: modeling the visual evolution of fashion trends with one-class collaborative filtering. In: WWW 2016, pp. 507–517 (2016)
12. Koren, Y., Bell, R., Volinsky, C.: Matrix factorization techniques for recommender systems. *Computer* **42**(8), 30–37 (2009)
13. Pan, W., Liu, N.N., Xiang, E.W., Yang, Q.: Transfer learning to predict missing ratings via heterogeneous user feedbacks. In: AAAI 2011, pp. 2318–2323 (2011)
14. Singh, A.P., Kumar, G., Gupta, R.: Relational learning via collective matrix factorization. In: KDD 2008, pp. 650–658 (2008)
15. Man, T., Shen, H., Jin, X., Cheng, X.: Cross-domain recommendation: an embedding and mapping approach. In: IJCAI 2017, pp. 2464–2470 (2017)
16. Xin, X., Liu, Z., Lin, C., Huang, H., Wei, X., Guo, P.: Cross-domain collaborative filtering with review text. In: IJCAI 2015, pp. 1827–1833 (2015)

17. Diao, Q., Qiu, M., Wu, C.Y., Smola, A.J., et al.: Jointly modeling aspects, ratings and sentiments for movie recommendation (JMARS). In: KDD 2014, pp. 193–202 (2014)
18. Zhang, Y., Lai, G., Zhang, M., et al.: Explicit factor models for explainable recommendation based on phrase-level sentiment analysis. In: SIGIR 2014, pp. 83–92 (2014)
19. Li, F., Wang, S., Liu, S., Zhang, M.: SUIF: a supervised user-item based topic model for sentiment analysis. In: AAAI 2014, pp. 1636–1642 (2014)
20. Baccianella, S., Esuli, A., Sebastiani, F.: SentiWordNet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining. In: International Conference on Language Resources and Evaluation, vol. 2542, pp. 83–90 (2010)



Fuzzy Gravitational Search Approach to a Hybrid Data Model Based Recommender System

Shruti Tomer¹ , Sushama Nagpal², Simran Kaur Bindra³ ,
and Vipra Goel⁴ 

¹ Samsung India Electronics Pvt Ltd, Noida, UP, India
shrutitomer01@yahoo.com

² Netaji Subhas Institute of Technology, Sector-3, Dwarka, Delhi 110078, India
sushmapriyadarshi@yahoo.com

³ Adobe Systems India Pvt Ltd, Noida, UP, India
salonibindra@gmail.com

⁴ Expedia Online Travel Svc Pvt Ltd, Gurugram, Haryana, India
vipral0goell1995@gmail.com

Abstract. In recent times, when the Internet is flooded with information, users get overwhelmed with the large amount of data and need some system to narrow down their choices. Recommender systems provide personalized suggestions to the users, giving them a better experience. Data Filtering methods along with many Computational Intelligence (CI) techniques have been used to build and optimize these systems. Here, we introduce a new Recommender System, based on Fuzzy Gravitational Search Algorithm using Hybrid Data Model (FGSA-HDM). FGSA-HDM uses a nature inspired heuristic technique, Gravitational Search Algorithm (GSA), to learn a user's preference and optimize weightage given to different features which define the user profile. Also, to incorporate the fuzziness of human nature, these features have been represented by Fuzzy sets. The proposed technique, FGSA-HDM, has shown better results than the previously implemented techniques - Pearson Correlation based Collaborative Filtering (PCF), Fuzzy Collaborative Filtering (FCF), Fuzzy Genetic Algorithm based Collaborative Filtering (FG-CF) and Fuzzy Particle Swarm Optimization based Collaborative Filtering (FPSO-CF).

Keywords: Recommender Systems · Computational intelligence techniques
Gravitational Search Algorithm · Fuzzy sets · Particle Swarm Optimization
Collaborative Filtering · Hybrid filtering

1 Introduction

Recommender Systems (RS) are Information Filtering Systems which aim to provide accurate and relevant recommendations to users. These systems have been gaining popularity in various domains like movies (Netflix, Moviepilot), music (Spotify, Pandora), news (Yahoo news), books (Kindle), jokes (Jester), e-commerce websites (Amazon, Flipkart), social media (Facebook, LinkedIn) and many more.

Recommender Systems use Data Filtering techniques to filter out content for their users. These techniques are widely categorized as Collaborative filtering (CF) [1], Content-based filtering (CBF) [2], Hybrid data filtering (HDF) [3] and Demographic data filtering (DMF) [4] techniques. Collaborative filtering creates a neighborhood of similar users, and suggests items which are highly rated by their neighbors. In Content based filtering, items which are like previous preferences of user are recommended. Demographic filtering takes user demographics like address, gender, age group, occupation etc. in consideration to infer recommendations. Each of the above-mentioned technique suffers from one or more drawbacks like Sparsity, Cold Start and Scalability. In this paper, we have used Hybrid data filtering technique as it is claimed to give better results because it overcomes some of the drawbacks of these individual approaches [5].

Traditionally, researchers working in the domain of Recommender Systems followed these practices:

1. Use of binary logic for user preference which does not consider the fuzzy behavior of human likes and dislikes.
2. Equal importance to every user profile feature like age, gender etc. while making recommendations, not taking into consideration that their inclination towards these features might differ individually.

Considering the above-mentioned aspects, recently, few authors explored various CI techniques based on Evolutionary Intelligence - Genetic Algorithm (GA) [6–8]; Swarm Intelligence - Ant Colony Optimization (ACO) [9], Particle Swarm Optimization (PSO) [10], Bee Colony Optimization (BCO) [11], Cuckoo Search Optimization [12]; Gravitational Search Algorithm (GSA) [13]; Artificial Neural Networks [14], Fuzzy Sets based RS [15] to provide more accurate and customized recommendations to users.

Here we introduce a technique, FGSA-HDM, which uses a hybridization of Computational Intelligence and Data Filtering techniques. The experimental results are based on the performance measured using the MovieLens dataset. Though the model is based on user similarity calculation using Collaborative filtering technique, Content and Demographic data are also incorporated in the user profile in the form of features. Figure 1 depicts the overall approach of our algorithm. The user profile features are organized into fuzzy sets, where the value present in each fuzzy set depicts participation of the feature in that fuzzy set. Also, each feature is considered to hold different weightage while calculating user-similarity for each user, during neighborhood selection. These feature weights are optimally calculated by implementing GSA for each user. GSA has been used because it has good convergence speed, better exploration and memory utilization as compared to other techniques, like PSO [16]. Also, it is a relatively less explored algorithm in the Recommender Systems' domain.

The performance of the approach proposed in this work is also compared with that of the approaches used earlier in this domain and was found to yield promising results based on the experiments conducted.

Rest of the paper is organized as follows: Sect. 2 illustrates the work that has already been done in Recommender Systems' domain. In Sect. 3, we discuss our proposed technique FGSA-HDM. Section 4 provides a description of GSA and how it is used to calculate the weightage of the features for each user profile. Section 5 describes the Experimental Settings. Finally, Sect. 6 concludes our attempt towards designing a better Recommender System and suggests areas for possible improvement.

2 Background

The techniques used for generating recommendations have evolved over the period, from using naïve techniques based on user-user similarity (Collaborative) or item-item similarity (Content Based) to using Computational Intelligence techniques along with data filtering techniques. Collaborative filtering is based on the principle of quantifying similarity between users. Various techniques have been used to calculate user similarity such as Pearson Correlation, Cosine Similarity, Euclidean distance, Squared Euclidean [17] etc. In these techniques, no attention has been given to the weightage each user-profile feature holds while calculating the similarity between users. Each feature has been assumed to hold equal preference for the user.

Vassiliou et al. used Artificial Neural Networks (ANN) to find similarity between the user profiles and items of interest to the user in [14]. This assisted in personalizing recommendations because of the self-learning capability of ANN.

Crisp values always hampered the process of getting more accurate user preferences. Suryavanshi et al. used Fuzzy Collaborative Filtering based RS for Web Personalization in [18]. Experimental results concluded that it worked better than Memory and Model based CF and was not only able to construct model in less time, but was also able to handle large datasets easily.

Along with data filtering techniques, Nature-inspired techniques, like Swarm Intelligence, were incorporated by various researchers to optimize weightage given to different user profile features. Genetic Algorithm (GA), an evolutionary search algorithm inspired from Darwinian Evolutionary theory [8], has been used to fine tune the profile matching algorithm within a RS introduced in [7]. Ant Colony Optimization (ACO) was proved to give better results over naïve CF, CBF, Demographic CF using Euclidean metric in [9]. Also, PSO performed better than naïve CF and GA [10].

A Bio-inspired technique called Cuckoo search has also been explored as an optimization algorithm with k-means clustering algorithm in [12] to generate a movie Recommender System.

Choudhary et al. tried a rather new heuristic optimization algorithm - Gravitational Search Algorithm (GSA) [16] for recommending jokes [13]. In this, GSA was found to perform better and was more memory efficient than PSO.

Hybridization of two or more CI techniques within an algorithm were found to improve the Recommender System's performance. User profiles were created in the form of fuzzy-feature sets and respective feature weights for each user were calculated using an optimization technique like PSO [19] or GA [20].

3 Proposed Approach for Generating Recommendations

The algorithm proposed in this paper incorporates a blend of different data filtering techniques making it a hybrid user data model and uses GSA for feature weight optimization. Further, the user profile features are distributed into fuzzy sets to have a more realistic approach. However, the combination of Fuzzy sets with GSA remains unexplored. Therefore, in this paper we have proposed a Hybrid User Model, using

GSA for feature weight optimization along with various data filtering techniques, and fuzzified user profile representation.

The task of generating recommendation consists of 3 stages [20]:

1. Collection of Data and User Profile creation.
2. Neighborhood Set generation.
3. Predicting User Ratings and making Recommendations.

3.1 Collection of Data and User Profile Creation

In this work, we have used MovieLens dataset. Data collected can be classified into age, gender (0/1 for male/female), occupation (from list of occupations as given in dataset) and 18 genre frequencies is created. For example, typical user profile of would look like [Age: 29, Occupation: 10, Gender: 1, Genre Frequencies: 010101010101101010].

Some of these features are then organized into fuzzy sets to take the human fuzzy behavior into account. For measuring each user's implicit interest in various movie genres, Genre Interestingness Measure (GIM) is calculated [20]. GIM of a user for a genre is measured based on how good a rating user has given to the movies of that genre. GIM for the 18 genres and age are further fuzzified into sets, whereas occupation and gender are taken to be fuzzy points of membership value of one.

3.2 Neighborhood Set Generation

Only the closest or the most similar users should be considered while making recommendations. In this paper, Euclidean distance metric has been used as the similarity measure.

To implement fuzzy logic in similarity calculation, we compute local fuzzy distance between users- u and v using the formula [20]:

$$LFD(u_i, v_i) = d(u_i, v_i) \times d'(u_i, v_i) \quad (1)$$

where $d(u_i, v_i)$ simply gives the absolute difference between the values of i^{th} feature of user profile. $d'(u_i, v_i)$ is the component incorporating fuzzy sets [20]:

$$d'(u_i, v_i) = \sqrt{\sum_{j=1}^m (u_{i,j} - v_{i,j})^2} \quad (2)$$

where $u_{i,j}$ represents value of j^{th} fuzzy set of the i^{th} feature for user u .

As a user might be influenced by each feature differently, we give different weights to different features while calculating the similarity measure. To implement this, we use Global Fuzzy Distance (GFD) between users- u and v , given as follows [19]:

$$GFD(u, v) = \sqrt{\sum_{f=1}^{21} w_f \times (LFD(u_f, v_f))^2} \quad (3)$$

where w_f represents weightage given to f^{th} feature by user u , calculated using GSA.

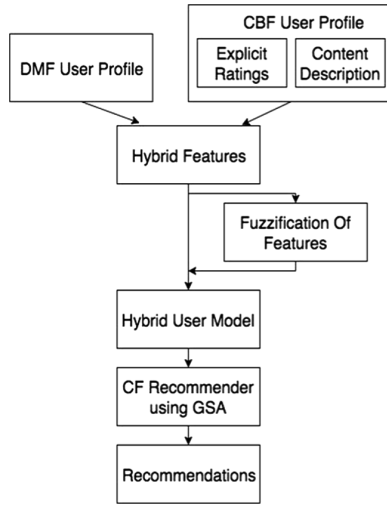


Fig. 1. Overview of our approach: FGSA-HDM

3.3 Predicting User Ratings and Making Recommendations

Ratings for each user u are predicted for the item i using the formula [20]:

$$pre_{u,i} = \bar{r}_u + k \sum_{u' \in S} d^l(u, v) \times (r_{v,i} - \bar{r}_v) \tag{4}$$

where S are the users in the neighborhood set who have rated the item i , $k = \frac{1}{\sum_{u' \in S} |d^l(u, v)|}$, acts as a normalizing factor, \bar{r}_v is the mean rating for user v and $r_{v,i}$ is the rating for item i given by that user.

Out of all the predicted items, we recommend only the highly rated ones and spare user of the effort of choosing from a bulk of recommendations.

4 Gravitational Search Algorithm

Gravitational Search Algorithm is a heuristic search technique based on the Law of Gravity. The agents participating in the algorithm are considered as objects having position, mass, velocity and acceleration. Each agent represents a possible solution and its mass gives its proximity to the optimal solution. Following the law of gravity, forces act on an agent because of all the other agents present in its vicinity causing change in its position, velocity and acceleration in each iteration. The agent depicting an optimal solution has a larger mass as compared to one that does not. A fitness function is chosen as per the problem domain and masses are calculated using this function after every iteration. The agent with greater mass moves lesser and pulls other agents towards it. Position of the agent with the greatest mass at the end of the iterations gives the optimal solution to the problem.

In our proposed model, GSA is used to calculate the feature weights w_f for each user and the following sections explains the detailed algorithm.

4.1 Fitness Function Used

Since we are attempting to predict ratings for a user as per his preferences, we choose fitness function to be the difference between predicted and actual ratings of an item for that user [20].

$$fitness = \frac{1}{t_R} \sum_{j=0}^{t_R} |r_j - pre_j| \quad (5)$$

Here r_j is the true rating of an item j for a user and t_R is the number of items in the training data set of users. Training and test sets are explained in Sect. 5. pre_j is the predicted rating for item j by a user, calculated using formula (4).

4.2 Algorithm

The algorithm is run for 30 iterations, as 30 iterations were found to give the best results in our experimental setup. In each iteration masses of all N agents (10 in our case) are updated using the fitness function. Since for an optimal solution we require the difference between the predicted and actual rating of an item to be minimum, ours is a minimization problem [16].

$$best(t) = \min_{j \in \{1, \dots, N\}} fit_j(t) \quad (6)$$

Here $fit_j(t)$ represents the fitness value of the j^{th} agent in t^{th} iteration. $best(t)$ gives the best fitness value in iteration t .

The algorithm is run for every user to find the optimal feature weights. There are 21 feature weights to be calculated. Therefore, the number of dimensions for the position vector of the agents is taken to be 21. The initial position P_i of each agent i is randomly initialized:

$$P_i = (d_1, d_2 \dots d_{21}) \quad (7)$$

here d_j is the position of i^{th} agent in the j^{th} dimension. Velocity and acceleration is initially kept 0 for all agents in every dimension.

The mass of each agent $m_i(t)$, the force applied by an agent j on agent i in d^{th} dimension and t^{th} iteration, $F_{ij}^d(t)$, and the gravitational constant, $G(t)$, are computed after each iteration using the following formula [16]:

$$m_i(t) = \frac{fit_i(t) - worst(t)}{best(t) - worst(t)} \quad (8)$$

$$M_i = \frac{m_i(t)}{\sum_{j=1}^N m_j(t)} \quad (9)$$

$$F_{ij}^d(t) = G(t) \times \frac{M_i(t) \times M_j(t)}{R_{ij} + \varepsilon} \times (s_j^d(t) - s_i^d(t)) \tag{10}$$

$$(t) = (-\alpha t/T) \tag{11}$$

where ε is a constant with small value which is randomly generated, R_{ij} is the GFD between agents i and j , and s represents their position vectors. G_o and α are initialized to 1 and $0.9 \times T$ where T is the total number of iterations. These parameters are gradually decreased to control the search space.

The total force on each agent in each direction is calculated using the formula [16]:

$$F_{ij}^d(t) = \sum_{j \in K_{best}, j \neq i} rand_j F_{ij}^d(t) \tag{12}$$

K_{best} is the set of best K agents i.e. agents with large masses, having the best fitness function values. Using K_{best} we are narrowing down the search space thus exploiting the area of optimal solution only. We started with K_{best} equal to swarm size and linearly decreased it by 1 after few iterations.

Acceleration and velocity of agents is updated using the following equations [16]:

$$a_i^d(t) = F_i^d(t)/M_i(t) \tag{13}$$

$$v_i^d(t+1) = rand_i \times v_i^d(t) + a_i^d(t) \tag{14}$$

$$s_i^d(t+1) = s_i^d(t) + v_i^d(t+1) \tag{15}$$

where $a_i^d(t)$, $v_i^d(t)$ and $s_i^d(t)$ are acceleration, velocity and position respectively of an agent i in d^{th} dimension at iteration t .

4.3 Termination Condition

The above discussed algorithm is repeated until a termination condition is met. There are various parameters that can serve as a termination condition like reaching a threshold fitness value, running the algorithm for a fixed number of iterations or allowing the agents to converge and stop when only one agent representing the global solution is left. In our approach, we terminate the algorithm after running 30 iterations. Position of the agent having the largest mass gives the solution i.e. feature weights for that user.

After obtaining feature weights, a neighborhood set is generated for each user using Global Fuzzy Distance (3). Top similar users constitute the neighborhood set and participate in the movie predictions for that user.

4.4 Comparison Between GSA and PSO

Although both GSA and PSO are population based stochastic methods, the basic strategy driving the motion of the particles differ and GSA is more promising than PSO as per the following reasons [13, 16]:

- In PSO, position and velocity are updated by only two factors i.e. *pbest* (the current agent's best position so far) and *gbest* (the best position amongst all the agents obtained so far). Whereas in GSA, these are calculated under the influence of force applied by all the agents present in its vicinity. Thus, GSA has a better exploration property and all the possible solutions are considered.
- PSO utilizes more memory by storing *pbest* and *gbest* of previous iterations, while in GSA only the current positions and forces of the agents play a role in the updating procedure.
- In GSA, the mass of each agent is made proportional to its fitness value in that iteration and slight modification in these masses may cause a notable change in global solution, whereas in PSO, the fitness value of all agent (*pbest*) in each iteration might not hold such importance to the global solution as they remain constant unless a higher *pbest* is achieved for that agent.
- GSA converges faster than PSO.

5 Experiment and Results

To give a practical exhibition of the proposed algorithm FGSA-HDM, the following experiments were conducted.

5.1 Experimental Setup

We used the MovieLens Dataset of 100K ratings available at grouplens.org. Out of 943, only those users were selected who have rated at least 60 movies. These users are further divided into active users and normal users, where predictions are made for the active users while normal users participate in neighborhood set generation Training set and test set comprise of 66% and 34% of user ratings respectively [19].

Results in this paper are shown for the combination of best tuned parameters listed in Table 1.

Also, 5-fold cross validation was performed with two variations:

- Variation 1: For each fold, 50 different active users were selected randomly and recommendations generated for them.
- Variation 2: 50 active users were randomly selected initially. In each fold, ratings of these active users were randomly divided into training and test datasets.

Table 1. Various parameters defined for algorithm

Parameter	Parameter value
Swarm size (agents participating in each iteration)	10
Neighborhood size (users similar to the active user)	30
Number of recommendations	≥ 3
Number of iterations	30

5.2 Evaluation Metrics

To test the ability of the proposed approach FGSA-HDM, and to compare its performance with others, we analyzed the results using Mean Absolute Error (MAE) and Coverage.

MAE measures the deflection of predicted ratings generated by the RS from the actual ratings mentioned in the dataset [21]. The $MAE(i)$ for active user u_i is given by the following formula:

$$MAE(i) = \frac{1}{t_i} \sum_{j=1}^{t_i} |pre_{ij} - r_{ij}| \tag{16}$$

where t_i is the cardinality of the test ratings set of the user u_i and r_{ij} is the actual rating given to j^{th} item by u_i . The total MAE over all the active users is:

$$MAE = \frac{1}{N} \sum_{i=1}^N MAE(i) \tag{17}$$

where N is the number of active users chosen at a time i.e. 50 in our case. Clearly, lower MAE corresponds to more accurate predictions of the given RS.

Coverage is defined as the ratio of items for which the RS can predict ratings over the items for which the predictions were expected for all the users. Lower Coverage indicates that the system is unable to make sufficient number of recommendations for the user [21]. Coverage is calculated by the formula:

$$Coverage = \frac{\sum_{i=1}^{T_n} q_i}{\sum_{i=1}^{T_n} t_i} \tag{18}$$

where q_i is the total number of items the algorithm could predict and t_i is the cardinality of the test ratings set of the user u_i .

5.3 Results

To demonstrate the recommendation ability of our approach FGSA-HDM, we compare its results with the previously implemented techniques in terms of MAE and Coverage. Table 2 shows MAE and Coverage for 5-fold experiments of our algorithm.

Table 2. MAE and coverage for 5 folds in FGSA-HDM for variation 1 and variation 2

Folds	MAE-1	Coverage-1	MAE-2	Coverage-2
1	0.6515	0.9636	0.6244	0.9675
2	0.6790	0.9658	0.6265	0.9670
3	0.6753	0.9669	0.6389	0.9676
4	0.6534	0.9616	0.6242	0.9667
5	0.6481	0.9678	0.6373	0.9622
Average	0.6614	0.9651	0.6282	0.9670

Table 3 shows a significant improvement in MAE and a considerable increase in Coverage for FGSA-HDM in comparison to other techniques - PCF, FCF, FG-CF, FPSO-CF [20]. This approach, thus, predicts ratings with greater accuracy and can give more recommendations to users. A Coverage value of 0.9651 indicates that the system can rate nearly all the items that it is expected to be predicted for a user. Figure 2 gives a pictorial representation of the comparison.

Table 3. Comparison of FGSA-HDM with PCF, FCF, FG-CF, FPSO-CF in terms of average MAE and coverage

Parameter	PCF	FCF	FG-CF	FPSO-CF	FGSA-HDM
MAE	0.8793	0.8262	0.8115	0.8006	0.6614
Coverage	0.8511	0.9451	0.9474	0.9549	0.9651

Further, to ensure validity of results we used this algorithm on EachMovie Dataset available at grouplens.org. The average MAE and Coverage came out to be 0.5244 and 0.9821 respectively, hence verifying the efficacy of the recommendations.

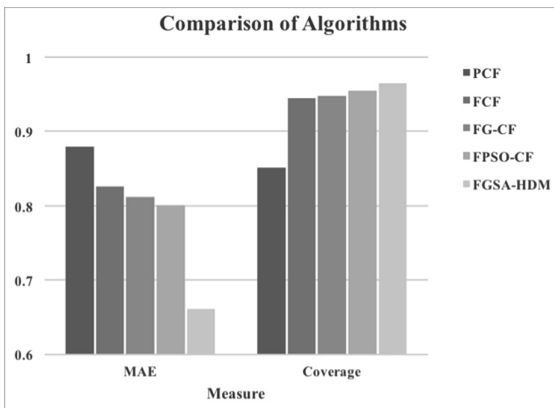


Fig. 2. Comparison of PCF, FCF, FG-CF, FPSO-CF, FGSA-HDM in terms of average MAE and coverage

6 Conclusion and Future Work

The proposed model FGSA-HDM uses explicit user ratings, demographic data as well as data derived from item (movie) description for creating a user profile. The demographic data serves the use cases where users with same age group, gender, occupation might have liking for similar items. Rather than taking crisp values of different features in the user’s profile, fuzzy sets for different features are created. Thus, fuzzification brings in the touch of ambiguous and imprecise human nature to our model and makes the user profile more realistic. Proposed FGSA-HDM RS performs better than the

traditional Recommender Systems. GSA has been used to find out the preference, in the form of weightage, given by each user to different features. Good exploration and exploitation abilities of GSA makes it a better algorithm for such optimization. Thus, GSA gives fast and accurate results. The proposed algorithm gives reduced Mean Absolute Error and increased Coverage than the previously used algorithms like PSO and GA.

The problems faced in Recommender System are that of Cold Start, Scalability and Data Sparsity. Cold Start is specific to new users and items. Using demographic data of the users reduces cold start problem in our proposed technique. In future, work can be done to reduce problems of data sparsity and scalability.

Also, for future experiments, trust value between different users can be added as a feature in the user profile to find a compact and efficient neighborhood set. Further, system can be improved by incorporating learning and renewal of knowledge and tastes of the user. Also, the given model can be extended to make Cross-Domain Recommendations.

References



1. Breese, J.S., Heckerman, D., Kladie, C.: Empirical analysis of predictive algorithms for collaborative filtering. In: Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence, pp. 43–52 (1998)
2. Lops, P., de Gemmis, M., Semeraro, G.: Content-based recommender systems: state of the art and trends. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) Recommender Systems Handbook, pp. 73–105. Springer, Boston, MA (2011). https://doi.org/10.1007/978-0-387-85820-3_3
3. Jain, K.N., Kumar, V., Kumar, P., Choudhury, T.: Movie recommendation system: hybrid information filtering system. In: Bhalla, S., Bhateja, V., Chandavale, A.A., Hiwale, A.S., Satapathy, S.C. (eds.) Intelligent Computing and Information and Communication. AISC, vol. 673, pp. 677–686. Springer, Singapore (2018). https://doi.org/10.1007/978-981-10-7245-1_66
4. Krulwich, B.: Lifestyle finder: intelligent user profiling using large-scale demographic data. *AI Mag.* **18**, 37 (1997)
5. Abbas, A., Zhang, L., Khan, S.U.: A survey on context-aware recommender systems based on computational intelligence techniques. *Computing* **97**(7), 667–690 (2015)
6. Bobadilla, J., Ortega, F., Hernando, A., Alcalá, J.: Improving collaborative filtering recommender system results and performance using genetic algorithms. *Knowl.-Based Syst.* **24**(8), 1310–1316 (2011)
7. Ujjin, S., Bentley, P.J.: Learning user preferences using evolution. In: Proceedings of the 4th Asia-Pacific Conference on Simulated Evolution and Learning, Singapore (2002)
8. Tang, K.S., Man, K.F., Kwong, S.: Genetic algorithms and their applications. *IEEE Sign. Process. Mag.* **13**(6), 22–37 (1996)
9. Sobacki, J., Tomczak, J.M.: Student courses recommendation using ant colony optimization. In: Nguyen, N.T., Le, M.T., Świątek, J. (eds.) ACIIDS 2010. LNCS (LNAI), vol. 5991, pp. 124–133. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-12101-2_14
10. Ujjin, S., Bentley, P.J.: Particle swarm optimization recommender system. In: 2003 IEEE Proceedings of the Swarm Intelligence Symposium, SIS 2003, pp. 124–131 (2003)

11. Bonabeau, M., Dorigo, G.: *Theraulaz: Swarm Intelligence*. Oxford University Press, Oxford (1997)
12. Katarya, R., Verma, O.P.: An effective collaborative movie recommender system with cuckoo search. *Egypt. Inf. J.* **18**(2), 105–112 (2017)
13. Choudhary, V., Mullick, D., Nagpal, S.: Gravitational search algorithm in recommendation systems. In: Tan, Y., Takagi, H., Shi, Y., Niu, B. (eds.) *ICSI 2017*. LNCS, vol. 10386, pp. 597–607. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-61833-3_63
14. Vassiliou, C., Stamoulis, D., Martakos, D., Athanassopoulou, S.: Recommender system framework combining neural networks and collaborative filtering. In: *Proceedings of the 5th WSEAS International Conference on Instrumentation, Measurement, Circuits and Systems*, Hangzhou, China, 16–18 April, pp. 285–290 (2006)
15. Zadeh, L.A.: Fuzzy sets. *Inf. Control* **8**, 338–353 (1965)
16. Rashedi, E., Nezamabadi-pour, H., Saryazdi, S.: GSA: a gravitational search algorithm. *Inf. Sci.* **179**(13), 2232–2248 (2009)
17. Cha, S.-H.: Comprehensive survey on distance/similarity measures between probability density functions. *Int. J. Math. Models Methods Appl. Sci.* **1**(4), 300–307 (2007)
18. Suryavanshi, N., Shiri, S.P.: Mudur: a fuzzy hybrid collaborative filtering technique for web personalization. In: *Proceedings of the Third Workshop on Intelligent Techniques for Web Personalization (ITWP 2005)*, Edinburgh, Scotland, UK (2005)
19. Wasid, M., Kant, V.: A particle swarm approach to collaborative filtering based recommender systems through fuzzy features. In: *Eleventh International Multi-Conference on Information Processing, (IMCIP-2015)*, Bengaluru, India (2015)
20. Al-Shamri, M.Y.H., Bharadwaj, K.K.: Fuzzy-genetic approach to recommender systems based on a novel hybrid user model. *Expert Syst. Appl.* **35**, 1386–1399 (2008)
21. Vozalis, E., Margaritis, K.G.: Analysis of recommender systems algorithms. In: *Proceedings of the Sixth Hellenic-European Conference on Computer Mathematics and its Applications (HERCMA)*, Athens, Greece (2003)

Probabilistic Models and Applications



Causal Discovery with Bayesian Networks Inductive Transfer

Haiyang Jia^{1,2,3} , Zuoxi Wu², Juan Chen^{1,2,3} , Bingguang Chen¹,
and Sicheng Yao¹

¹ College of Computer Science and Technology, Jilin University,
Changchun 130012, China
chenjuan@jlu.edu.cn

² College of Software, Jilin University, Changchun 130012, China

³ Key Laboratory of Symbolic Computation and Knowledge Engineering
of Ministry of Education, Jilin University, Changchun 130012, China

Abstract. Bayesian networks (BNs) is a dominate model for representing causal knowledge with uncertainty. Causal discovery with BNs requiring large amount of training data for learning BNs structure. When confronted with small sample scenario the learning task is a big challenge. Transfer learning motivated by the fact that people can intelligently apply knowledge learned previously to solve new problems faster or with better solutions, the paper defines a transferable conditional independence test formula which exploit the knowledge accumulated from data in auxiliary domains to facilitate learning task in the target domain, a BNs inductive transfer algorithm were proposed, which learning the Markov equivalence class of BNs. Empirical experiment was deployed, the results demonstrate the effectiveness of the inductive transfer.

Keywords: Bayesian network · Inductive transfer · Causal discovery
Structure learning

1 Introduction

Bayesian Networks (BNs) is a dominate model for representing causal knowledge with uncertainty. BNs has been researched for decades [1] and applied in many fields, such as computational biology (Predicting Protein-Protein Interaction [2], gene regulatory network [3]), medical (healthcare [4], disease progression mechanisms analysis [5]), legal (legal decision making [6]) and agriculture [7]. Causal discovery based on one of the most challenging issue: BNs structure learning. Although, many approaches have been proposed to increase the accuracy and efficiency of BNs learning, but most of them depend on the huge amount of training data. In some domains, such as biology and agriculture, both the training data and expert knowledge is very rare and precious. In the small sample scenario, causal discovery with BNs structure learning cannot be implemented with traditional approaches.

When confronted with small sample issue, the learning task is a big challenge, a new learning schema was developed that using data from related tasks to alleviate the issue [8–10]. This paper proposed a BNs inductive transfer approach for causal discovery.

The remainder of this paper is organized as following: Sect. 2 introduces background knowledge; Sect. 3 describes the algorithm in detail; Sect. 4 describes the empirical experiment and finally Sect. 5 draw the conclusion.

2 Research Background

Notation: capital letter X, Y, Z notate a random variable; lowercase x, y, z notate the value of a random variable; bold capital letter $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$ notate a set of random variable. (v_1, v_2) notate a undirected edge between v_1 and v_2 , $\langle v_1, v_2 \rangle$ notate a directed edge from v_1 to v_2 . $|E|$ is the number of element in set E .

2.1 Bayesian Network Learning

A BNs is a concise representation of joint probability distribution on a set of random variables [11]. A BNs, represent the joint probability distribution of a set of random variables $\mathbf{X} = (X_1, X_2, \dots, X_n)$, consists of two parts $BNs = \langle G, \theta \rangle$: G is a directed acyclic graph(DAG), each node corresponds to a random variable in \mathbf{X} , G encodes the independencies of the probability distribution. θ is conditional probability table (CPT), encoding the conditional distributions of each family (a node and its parent node), $\theta = \{p(X_i|\pi_{X_i})|1 \leq i \leq n\}$ (π_{X_i} is the parent nodes of X_i).

Briefly, the joint probability distribution represented by BNs is:

$$P(X_1, X_2, \dots, X_n) = \prod_{i=1}^n P(X_i|\pi_{X_i}) \quad (1)$$

BNs learning includes two aspects: learning the structure and learning the parameters. When the structure is given, parameter learning algorithm learning the CPTs from data with Maximum likelihood estimation or Bayesian estimation [12].

BNs structure learning is the key component of BNs learning, there are mainly two categories of BNs structural learning algorithm:

- (1) score-based approach, which define the learning task as an optimization problem. Using a scoring function to evaluates the structure G given a data set D , mainstream score criteria include AIC (Akaike information criterion) [13], BIC (Bayesian information criterion) [14], MDL (minimum description length) [15] and BDe (Bayesian Dirichlet equivalent) [16];
- (2) constraint-based approach, which define the learning task as constraint satisfaction problem. Applying conditional independent (CI) test to find the conditional independence relationships in data D , then construct a BNs satisfied such constraint [17].

The constraint-based are usually much more efficient when the number of variables is large. However, when the sample size is small, and the data is noisy, the scoring-based algorithms can often more accurate and robust. Hybrid technique that combines a mixed approach of score-based approach with constraint-based approach [18, 19].

There is Markov equivalence among BNs, essential graph is a graphical representation of Markov equivalence classes (MEC), Essential graph provide a more general graphical model for knowledge representation.

Definition 1. Two BNs, BN_1 and BN_2 , are call Markov equivalent iff:

- (1) BN_1 and BN_2 have same random variable set \mathbf{X} ;
- (2) The probability distribution that can be represented by BN_1 can also be represented by BN_2 , Vice-versa.

BNs Markov equivalent can be determined by the following procedure [20].

Definition 2. V_i, V_j is a V structure at V_k in the DAG $G = (V, E)$ iff.:

$V_i, V_j, V_k \in V, \langle V_i, V_k \rangle \in E, \langle V_j, V_k \rangle \in E, \langle V_i, V_j \rangle, \langle V_j, V_i \rangle \notin E$, short for $VS(i, j|k)$.

Definition 3. The *skeleton* of a DAG is the undirected graph resulting from ignoring the directionality of every edges.

Theorem 1. BN_{S_1} and BN_{S_2} are *Markov Equivalence* iff. BN_{S_1} and BN_{S_2} have the same skeleton and same V structure [20, 21], sort for $ME (BN_{S_1}, BN_{S_2})$. All DAGs that equivalent to each other is a *Markov equivalence class*, short for MEC.

Definition 4. *Essential Graph* (short for EG) is an acyclic partially directed graph (PDAG) which contains both directed and undirected edges (acyclic in the sense that it contains no directed cycles). The direction of each directed edge was identified as following:

$ME (BN_{S_i}, BN_{S_j}), \forall i, j \in [1, k], G_i = \langle V_i, E_i \rangle 1 \leq i \leq k, k = |MEC| EG = \langle V, E \rangle$, E is a set of direct and undirected edge, $V = V_1 \dots = V_k; E = \bigcup_{i=1}^k E_i$, if $\langle v_i, v_j \rangle \in E \wedge \langle v_j, v_i \rangle \in E$ it means $(v_i, v_j) \in E$.

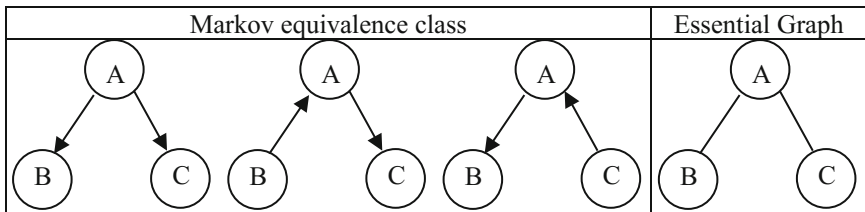


Fig. 1. Example for *Markov equivalence class* and *essential graph*

2.2 Transfer Learning and Inductive Transfer

The study of transfer learning (TL) is motivated by the fact that people can intelligently apply knowledge learned previously to solve new problems faster or with better solutions [8]. In contrast to classical machine learning methods, TL exploit the knowledge accumulated from data in auxiliary domains to facilitate predictive modeling consisting of different data patterns in the current domain [22]. TL allows the domains and distributions used in training and testing to be different. This paper following the definition and categorization in Pan and Yang’s work [8] (Fig. 1):

Definition 5. Domain D consists of two components: A feature space X , a marginal distribution $P(\mathbf{X})$, $\mathbf{X} = \{X_1, X_2, \dots, X_n\} \in X$. $D = \{X, P(\mathbf{X})\}$.

In general, if two domains are different, then they may have different feature spaces or different marginal distributions.

Definition 6. Task T , given a specific domain $D = \{X, P(\mathbf{X})\}$, task consists of two components: label space Y and an objective predictive function $f(\cdot)$. for each X_i in the domain, $f(\cdot)$ predict its corresponding label Y_i , where $Y \in_i Y$. $T = \{Y, f(\cdot)\}$.

In general, if two tasks are different, then they may have different label spaces or different conditional distributions $P(Y|X)$, where $Y = \{Y_1, Y_2, \dots, Y_n\} \in Y$. From a probabilistic view $f(X)$ can be written as $P(Y|X)$. BNs learning joint distribution, that means both marginal distribution and conditional distribution will be trained, so $f(X)$ can be written as $P(Y, X) = P(Y|X) P(X)$.

Definition 7. Transfer Learning, given a source domain $D^S = \{X^S, P(X^S)\}$, $X^S = \{X_1^S, X_2^S, \dots, X_n^S\} \in X^S$ and learning task T^S , a target domain $D^T = \{X^T, P(X^T)\}$, $X^T = \{X_1^T, X_2^T, \dots, X_n^T\} \in X^T$ and learning task T^T . TL aims to help improve the learning of the target predictive function $f^T(\cdot)$ in D^T using the knowledge in $D^S \leftarrow$ and $T^S \leftarrow$, where $D^S \neq D^T \leftarrow$, or/and $T^S \neq T^T$.

TL was categorized under three sub-settings, Inductive TL, transductive TL and unsupervised TL, based on different situations between the source and target domains and tasks (Table 1).

Table 1. Transfer learning categorization

Learning settings		D^S and D^T	T^S and T^T
Traditional machine learning		$D^S = D^T$	$T^S = T^T$
Transfer learning	Inductive TL/	$D^S = D^T$	$T^S \neq T^T$
	unsupervised TL	$D^S \neq D^T$	$T^S \neq T^T$
	Transductive TL	$D^S \neq D^T$	$T^S = T^T$

In the inductive transfer learning setting, the target task is different from the source task, no matter the source and target domains are the same or not.

2.3 Related Work

In the context of TL in BNs, Oyen given the formulations for TL of BNs [23], Mizil present a multi-task framework of structure transfer, it assumes all sources are equally related and learns the parameters for each task independently [24]. Luis present a constrain-based approach, assumes every source is both relevant and equally related [25]. Oyen considers multi-task structure learning that structural bias can be incorporated into the order-conditioned network discovery formulation [26]. Oates present a integer linear programming approach for joint estimation over multiple structure [27]. Fiedler present a temporal nodes BNs transfer learning algorithm [28]. Zhou present a BNs parameter TL algorithm [29].

3 BNs Inductive Transfer

3.1 Formulation and Assumption

Given a set of source domain $D^S = \{D^{S,i} \mid i \in [1, K]\}$ and one target domain D^T . For BNs learning, each domain has its own training data, $D^{S,i}$ and D^T , the training data comes from different but similar distribution. The paper assume that all domains have the same set of random variables $\mathbf{X} = (X_1, X_2, \dots, X_n)$ that corresponding to nodes of the BNs, each node have same set of value in all domains. For coherence, target domain defined as number 0 domain, so all domain represented as $D = \{D^i \mid i \in [0, K]\}$. The goal of BNs inductive transfer structure learning is to get a best estimated BNs structure $\hat{G}^0 = \arg \max_G P(G^0 \mid D^i)$. $i \in [0, K]$.

3.2 Transfer Conditional Independence Test

Causal discovery with Bayesian networks is the process of BNs structure learning, the inductive TL leverage knowledge among a set of related source tasks (BNs structure learning) by applying a bias toward learning similar target BNs structure. If BNs structure was transferred as a directed acyclic graph (DAG), that may cause a misleading MEC bias, that is the MEC with more DAGs will have higher prior probability.

For example, in Fig. 2, assume the prior probability for the 8 DAGs is equal distribution, then the implicit prior probability of the three MEC, represented by essential graph, is biased by the number of DAGs in same MEC.

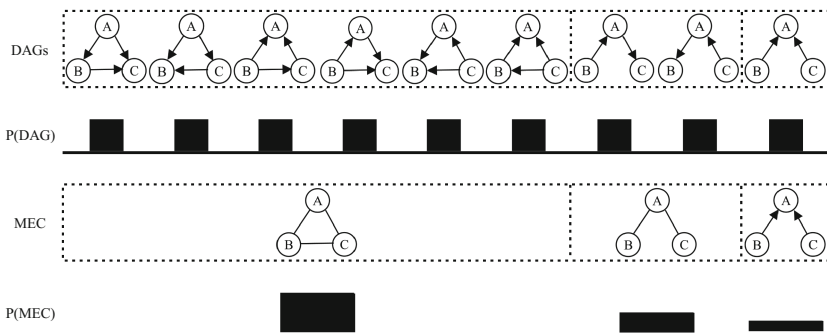


Fig. 2. An example of MEC bias, the structures in a dot rectangle belong to same MEC.

Therefore, transfer MEC is more prefer than DAG. Based on the definition of BNs, if A is independent of B, notated as $Ind(A, B) = 1$, then there is no edge between A and B, if A is dependent with B, notated as $Ind(A, B) = 0$, we cannot get the conclusion that there is an edge between A and B. Because there may be an active path between A and B that make A depend on B. There is an edge between A and B in the skeleton iff. $Ind(A, B|C) = 0$, for all possible C [30].

There are many metrics to test the independence, we employed the Mutual information to perform conditional independence test (CI test):

$$I(A, B) = \sum_{a,b} P(a, b) \log \frac{P(a, b)}{P(a)P(b)}$$

$$I(A, B|C) = \sum_{a,b,c} P(a, b, c) \log \frac{P(a, b|c)}{P(a|c)P(b|c)}$$

It is supposed A is (conditional) independent of B when $I < \epsilon$. For BNs inductive transfer, the transferable CI test employed the formulation below:

$$Ind^0(A, B|C) = \varphi\left(\sum_{i \in [0, K]} w_i Ind^i(A, B|Ds^i, C) - \lambda\varphi(C_{size} - \tau)\right)$$

$$w_i = \frac{\varphi(|\varphi(\tau - C_{size}) - \varphi(i)|) + (1 - \varphi(\tau - C_{size}))(1 - 2\varphi(i))(1 - C_{size}/n)}{M}$$

$$M = \varphi(\tau - C_{size}) + (1 - \varphi(\tau - C_{size}))(1 + \varphi(i)(K - 2)C_{size}/n)$$

$$\varphi(x) = \begin{cases} 0, & x \leq 0 \\ 1, & x > 0 \end{cases} \quad i \in [1, K]$$

Where w_i is the weight for each domain, $w_i \in [0, 1]$, $i \in [1, K]$, for each C_{size} $\sum_{i=0}^K w_i = 1$. If $w_0 = 1, w_i = 0, i \in [1, K]$, the learning is traditional single task learning in target domain. Ds^i is the training data set from domain i . n is the total number of nodes. C is a set of nodes, C_{size} is the number of nodes in C , $C_{size} \in [0, n-2]$. λ is the threshold that we will accept $Ind^0(A, B|C)$. M is a constant to normalize the weight. While increasing C_{size} , to keep the accuracy of CI test, the amount of the training data needed increase exponentially. As the target domain only has small amount of data, when C_{size} is less than τ , $Ind^0(A, B|C)$ use single task learning approach with Ds^0 only; when C_{size} is larger than τ , information from source domain will be transferred, while

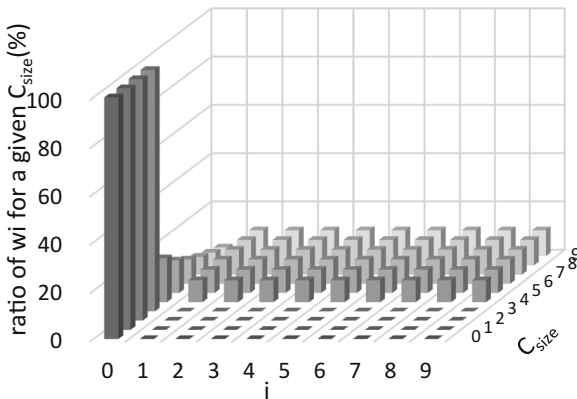


Fig. 3. The distribution of transfer weight w_i

C_{size} increased weight for source domain also increased. The Fig. 3 gives an example, $\tau = 4$, $n = 12$, illustrate the distribution of transfer weight, as the weight is less than 1, for the conciseness, the values of each w_i was timed by 100, then correspond to the percentage the information from domain i will take effect for a given C_{size} .

3.3 BNs Skeleton Discovery

With the transfer CI test defined above, the following algorithm, a variation of PC algorithm [30], performed the BNs skeleton transfer discover and edge orient, the algorithm ST learning the skeleton of target BNs⁰:

Algorithm ST (Data^{0...K}, SK⁰, CV, IV)

/* Using transferable CI test for skeleton transfer, input training Data^{0...K} for each domain, variable dimension for each Data set is $n \mathbf{V} = \{V_1, V_2, \dots, V_n\}$. Output the target skeleton SK⁰ for target domain, and a set CV for can be a V-structure, IV for impossible be a V-structure */

```

1 :   INIT [ initiate SK0 with a complete undirect graph; CV and IV as empty set]
2 :   SK0 = {(Vi, Vj) |  $\forall i, j \in [1, n], i \neq j$ }.
3 :   CV =  $\emptyset$ . IV =  $\emptyset$ .
4 :   TCIT [performing transferable CI test, determine whether Ind0(Vi, Vj | C) for  $\forall i, j \in$ 
5 :     [1, n], |C|=Csize  $\in [0, n - 2]$ , if true eliminate (Vi, Vj)]
6 :   FOR  $\forall C_{size} \in [0, n - 2]$ 
7 :     FOR  $\forall i, j \in [1, n], i \neq j \wedge (V_i, V_j) \in SK^0$ 
8 :       FOR  $\forall C = \{V_k | k \in [1, n], k \neq j, k \neq i \wedge (V_i, V_k) \in SK^0\}$ 
9 :          $\wedge |C| = C_{size} \in [0, n - 2]$ 
10 :        IF Ind0(Vi, Vj | C). SK0 = SK0 - (Vi, Vj).
11 :    CV [ find the V-structure in SK0]
12 :    FOR  $\forall (V_i, V_j, V_k), (V_i, V_k) \in SK^0, (V_i, V_j) \in SK^0, (V_j, V_k) \notin SK^0$ 
13 :      IF Ind(Vj, Vk | D0).
14 :        CV = CV + (Vi, Vj, Vk).
15 :      ELSE IF Ind(Vj, Vk | Vi, D0)
16 :        IV = IV + (Vi, Vj, Vk). ■

```

The output SK⁰ present the skeleton, CV and IV describe all V-structure, with Theorem 1, the algorithm ST output encode a specific MEC, graphical model for MEC is the PDAG. Compare with DAG, PDAG is more general. Generate all DAG that consist with PDAG is relatively straightforward, the algorithm given by Spirtes [30].

4 Experiments

Artificial domain and data was simulated: three BNs were selected form BNs repository as the benchmark, “CHILD”, “ALARM” and “HEPAR2”, these BNs set as target domain, denoted as BN_c⁰, BN_a⁰, BN_h⁰ respectively, presented in Fig. 4. Randomly

change the structures of each BNs^0 to generate source domains. Three structure change operators were applied, delete insert and reverse, the ratio of operator used to all edges in G^0 is 5%, 10%, 20%, 40%, 60%, 80%. K duplications were generated at each ratio, the generated BNs is used as source domains: $\{BNs_{c,r}^i, BNs_{a,r}^i, BNs_{h,r}^i\}$, r is the changing ratio $r \in \{5, 10, 20, 40, 60, 80\}$, i is the order number for the duplications $i \in [1, K]$. To denote the target BNs in a consist way with source BNs, $BNs_{c,r}^0 = BNs_c^0 \forall r$, then all BNs can be denoted in a more concise way:

$$\{BNs_{c,r}^i, BNs_{a,r}^i, BNs_{h,r}^i\}, r \in \{5, 10, 20, 40, 60, 80\}, i \in [0, K]$$

In the following paper, the first subscript will be omitted, means arbitrary r value, $BNs_r^i = BNs_{j,r}^i$, for $\forall j$.

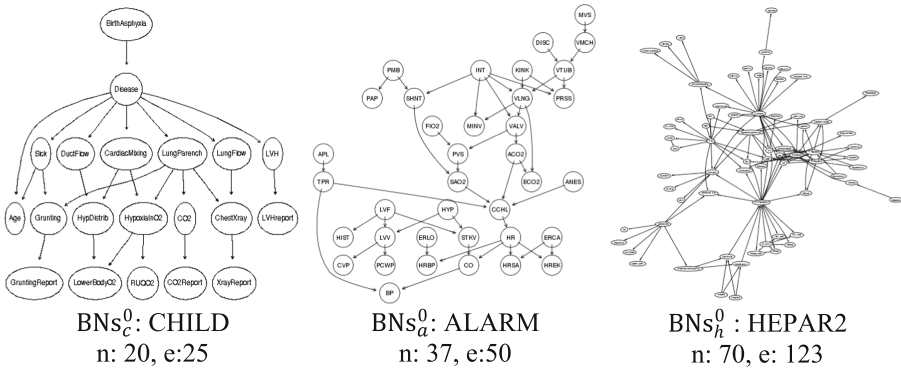


Fig. 4. Benchmark BNs from Bayesian Network Repository, n is the number of nodes e is the number of edges (<http://www.bnlearn.com/bnrepository/>)

Training data D_r^i was generated with BNs_r^i , when $i = 0$, the data size was limited. So $|D_r^0| \ll |D_r^i|, i > 0$. To illustrate the efficiency of inductive transfer, 4 approaches were deployed to learn the target BNs structure:

1. $G1(\widehat{BNs}^0) = \arg \max_G P(G^0 | D^0)$, a single task learning without the source domain data;
2. $G2(\widehat{BNs}_r^0) = \arg \max_G P(G^0 | mix(D_r^i)) i \in [0, K]$, simply mix the data from source domain and target domain together, then learning the BNs structure;
3. $G3(\widehat{BNs}_r^0) = \arg \max_G P(G^0 | D_r^i) i \in [0, K]$, transfer learning the target domain with K source domain, all source domain have the same change ratio;
4. $G4(\widehat{BNs}_r^0) = \arg \max_G P(G^0 | D_r^i) i \in [0, K], r \in \{5, 10, 20, 40, 60, 80\}$, all source domains at different changing ratio used together for inductive transfer.

Compare the underlying true structure of BNs^0 with the learning result to find differences. The error edges were classified into three groups: added edges, dropped

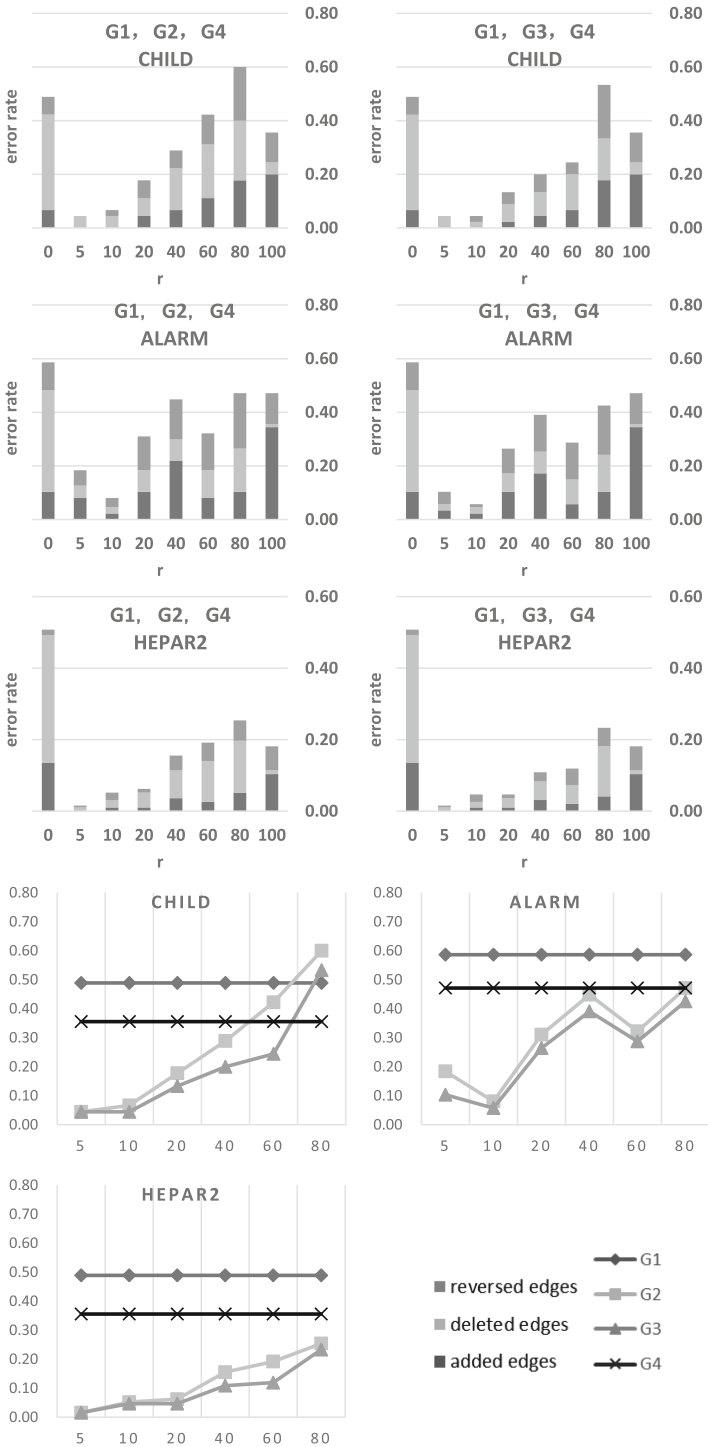


Fig. 5. The learning error rate of each experiment.

edges and reversed edges, to make the results comparable between different BNs the number of error edges divided by $(n^0 + e^0)^2$, where n^0 and e^0 is the number of nodes and edges in BNs⁰ make the all results presented in a concise way, G1 and G4 treat same as an instance of G2/G3 when $r = 0$ and $r = 100$, the results represent in Fig. 5.

G1 have the highest error rate, that means traditional single task learning cannot deal with small sample case well. While the changing rate increased, the error rate of G2 and G3 both increased, as the source and target being more and more different, transfer learning will take less effect, in some case negative transfer happened, learning results are worse than the single task learning. In all case G3, the inductive transfer approach, have the lowest error rate.

5 Conclusion

BNs is a dominate model for representing causal knowledge with uncertainty, causal discovery with BNs requiring large amount of training data, when confronted with small sample the learning task is big challenge. The paper defines a transferable conditional independence test formula which exploit the knowledge accumulated from data in auxiliary domains to facilitate learning task in the target domain, a BNs inductive transfer algorithm were proposed, which learning the Markov equivalence class of BNs. Empirical experiment was deployed, the results demonstrate the effectiveness of the inductive transfer. The work is based on the domains homogeneous assumption, heterogeneous transfer is the future work.

Acknowledgements. This paper is supported by National Natural Science Foundation of China under Grant Nos. 61502198, 61472161, 61402195, 61103091 and the Science and Technology Development Plan of Jilin Province under Grant No. 20160520099JH, 20150101051JC.

References

1. Heckerman, D.: A Bayesian approach to learning causal networks. In: Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence. Morgan Kaufmann Publishers Inc. (1995)
2. Jansen, R., et al.: A Bayesian networks approach for predicting protein-protein interactions from genomic data. *Science* **302**(5644), 449–453 (2003)
3. Lo, L.Y., et al.: High-order dynamic Bayesian network learning with hidden common causes for causal gene regulatory network. *BMC Bioinf.* **16**, 395 (2015)
4. Velikova, M., et al.: Exploiting causal functional relationships in Bayesian network modelling for personalised healthcare. *Int. J. Approx. Reasoning* **55**(1), 59–73 (2014)
5. Koch, D., Eisinger, R.S., Gebharter, A.: A causal Bayesian network model of disease progression mechanisms in chronic myeloid leukemia. *J. Theor. Biol.* **433**, 94–105 (2017)
6. Thagard, P.: Causal inference in legal decision making: explanatory coherence vs. Bayesian networks. *Appl. Artif. Intell.* **18**(3–4), 231–249 (2004)
7. Drury, B., et al.: A survey of the applications of Bayesian networks in agriculture. *Eng. Appl. Artif. Intell.* **65**, 29–42 (2017)

8. Pan, S.J., Yang, Q.: A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* **22**(10), 1345–1359 (2010)
9. Silver, D., Bennett, K.: Guest editor's introduction: special issue on inductive transfer learning. *Mach. Learn.* **73**(3), 215–220 (2008)
10. Caruana, R.: Multitask learning. *Mach. Learn.* **28**(1), 41–75 (1997)
11. Pearl, J.: *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, pp. 117–133. Morgan Kaufmann, San Mateo (1988)
12. Yao, T.S., Choi, A., Darwiche, A.: Learning Bayesian network parameters under equivalence constraints. *Artif. Intell.* **244**, 239–257 (2017)
13. Akaike, H.: A new look at the statistical model identification. *IEEE Trans. Autom. Control* **19**(6), 716–723 (1974)
14. Gideon, S.: Estimating the dimension of a model. *Ann. Statist.* **6**(2), 461–464 (1978)
15. Lam, W., Bacchus, F.: Learning Bayesian belief networks: an approach based on the MDL principle. *Comput. Intell.* **10**, 269–293 (1994)
16. Heckerman, D., Shachter, R.: Decision-theoretic foundations for causal reasoning. *J. Artif. Intell. Res.* **3**, 405–430 (1995)
17. Cheng, J., et al.: Learning Bayesian networks from data: an information-theory based approach. *Artif. Intell.* **137**(1–2), 43–90 (2002)
18. Tsamardinos, I., Brown, L.E., Aliferis, C.F.: The max-min hill-climbing Bayesian network structure learning algorithm. *Mach. Learn.* **65**(1), 31–78 (2006)
19. Jia, H., et al.: A Hybrid method for orienting edges of Bayesian network. *Acta Electronica Sinica* **37**(8), 1842–1847 (2009)
20. Verma, T., Pearl, J.: An algorithm for deciding if a set of observed independencies has a causal explanation. In: *Uncertainty in Artificial Intelligence Proceedings of the Eighth Conference*. Morgan Kaufman, San Francisco (1992)
21. Thomas, V., Judea, P.: Equivalence and synthesis of causal models. In: *Proceedings of the Sixth Annual Conference on Uncertainty in Artificial Intelligence*. Elsevier Science Inc. (1991)
22. Lu, J., et al.: Transfer learning using computational intelligence: a survey. *Knowl.-Based Syst.* **80**, 14–23 (2015)
23. Oyen, D., Lane, T.: Leveraging domain knowledge in multitask Bayesian network structure learning. In: *Twenty-Sixth AAAI Conference on Artificial Intelligence* (2012)
24. Alexandru, N.M., Caruana, R.: Inductive transfer for Bayesian network structure learning. In: *11th AISTATS* (2007)
25. Luis, R., Sucar, L., Morales, E.: Inductive transfer for learning Bayesian networks. *Mach. Learn.* **79**(1–2), 227–255 (2010)
26. Oyen, D., Lane, T.: Transfer learning for Bayesian discovery of multiple Bayesian networks. *Knowl. Inf. Syst.* **43**(1), 1–28 (2015)
27. Oates, C.J., et al.: Exact estimation of multiple directed acyclic graphs. *Statist. Comput.* **26**(4), 797–811 (2016)
28. Fiedler, L.J., Sucar, L.E., Morales, E.F.: Transfer learning for temporal nodes Bayesian networks. *Appl. Intell.* **43**(3), 578–597 (2015)
29. Zhou, Y., Hospedales, T.M., Fenton, N.: When and where to transfer for Bayesian network parameter learning. *Expert Syst. Appl.* **55**, 361–373 (2016)
30. Spirtes, P., Glymour, C., Scheines, R.: *Causation, Prediction, and Search*. Springer, New York (1993). <https://doi.org/10.1007/978-1-4612-2748-9>



Robust Detection of Communities with Multi-semantics in Large Attributed Networks

Di Jin^{1(✉)}, Ziyang Liu^{1(✉)}, Dongxiao He¹, Bogdan Gabrys²,
and Katarzyna Musiał²

¹ School of Computer Science and Technology, Tianjin University,
Tianjin 300350, China

{jindi, liuziyang, hedongxiao}@tju.edu.cn

² Advanced Analytics Institute, School of Software,

Faculty of Engineering and IT, University of Technology Sydney,

PO Box 123 Broadway, Ultimo, NSW 2007, Australia

{Bogdan.Gabrys, Katarzyna.Musial-Gabrys}@uts.edu.au

Abstract. In this paper, we are interested in how to explore and utilize the relationship between network communities and semantic topics in order to find the strong explanatory communities robustly. First, the relationship between communities and topics displays different situations. For example, from the viewpoint of semantic mapping, their relationship can be one-to-one, one-to-many or many-to-one. But from the standpoint of underlying community structures, the relationship can be consistent, partially consistent or completely inconsistent. Second, it will be helpful to not only find communities more precise but also reveal the communities' semantics that shows the relationship between communities and topics. To better describe this relationship, we introduce the transition probability which is an important concept in Markov chain into a well-designed nonnegative matrix factorization framework. This new transition probability matrix with a suitable prior which plays the role of depicting the relationship between communities and topics can perform well in this task. To illustrate the effectiveness of the proposed new approach, we conduct some experiments on both synthetic and real networks. The results show that our new method is superior to baselines in accuracy. We finally conduct a case study analysis to validate the new method's strong interpretability to detected communities.

Keywords: Community detection · Social networks · Semantics
Transition probability · Nonnegative matrix factorization

1 Introduction

Network science is a modern and significant discipline in many fields, such as social and computer science. Networks, consisting of nodes and edges which connect a pair of nodes, always occur in a variety of contexts [1]. The real-world networks usually share the same characteristic: they exhibit strong community structure. The property of community structure is: in which network nodes are joined together in tightly knit groups, between which there are only looser connections [2]. For example, in

Facebook, users who have consistent interests often gather together and form a community but there are only few connections between such communities. Community structure reveals the fundamental functional modules of a network and enables us to better understand the interactive behavior of the network.

Community detection has developed rapidly in recent years and various community detection methods, which mainly focus on network topology, have been proposed, e.g., the agglomerative or divisive algorithms [3], modularity optimization based methods [4], and spectral algorithms [5]. Further, it is well known that a node may belong to multiple communities (i.e. overlapping community). As a result, lots of methods were developed to detect overlapping community, such as k -clique community detection algorithms [6], local expansion and optimization algorithms [7] and probabilistic model-based algorithms [8]. Except for network topology, node attributes or link attributes are also taken into account when discovering communities [9–11]. In addition to improving community detection, researchers have realized that community detection should not only find community structure but also describe communities semantically by the use of abundant verbal information in the textual content. These descriptions can reveal why some nodes form a community and enable people to better understand the functions or meanings of communities, and in a way, this has much more practical value in real-world applications. Some methods have been proposed which combine topology and content information and give reasonable and interpretable communities [12, 13].

However, some problems still occur and need to be solved when network topology and node contents are integrated. One of the most important issues is the mismatch problem of topology and content. Traditional methods [12–14] typically assume that the network topology and node contents share the same community membership, but in many real social networks, this assumption does not always hold. For example, in a Twitter network, social links usually directly reflect which users gather into a community, while users may generate diverse and disordered content information. Thus, the community membership derived by network topology probably differs from the cluster membership derived by node contents.

For the above problem, it is necessary to extract useful content information to assist topology information in detecting more actual and accurate communities. In this paper, we propose a new generative model different from the traditional generative model and design a new community detection method, referred to as Robust and Strong Explanatory Community Detection (RSECD). To be specific, based on nonnegative matrix factorization (NMF), we are able to obtain the community membership matrix for network topology and cluster membership matrix for node contents. More importantly, there exists some implicit relation between network communities and content clusters, thus we introduce a transition probability matrix to depict it. As a result, even though the content information does not match with topology information, our method can still obtain accurate detection results by using the transition matrix with a suitable prior. At last, we put network topology, node content and transition matrix into a unified NMF framework, and optimize them altogether by designing effective updating rules in order to achieve an integral balance of them.

In the experiments, we use artificial networks to analyze the parameter in the objective function and to demonstrate the effectiveness and robustness of our approach.

Next, we conduct experiments on seven real-world network datasets and compare RSECD with eight baseline methods in terms of both disjoint community and overlapping community evaluation metrics. Experimental results show that RSECD can significantly improve the performance in all comparisons, which further illustrates our approach's robustness. And finally, in order to verify that RSECD is strong-explanatory to communities, we use a case study on a musical social network to semantically explain the hidden meanings of some topics and tell the 'true stories' behind communities.

2 Related Work

Various community detection methods, which only take the network topology into account, have been proposed. For example, hierarchical clustering methods [3] which include agglomerative and divisive hierarchical algorithms. Optimal modularity approaches (such as spectrum optimization method [5]) can find communities by the use of modularity optimization. Another approach [4] applies modularity into graphs of different networks by correcting modularity, such as symbolized networks. By mapping a network into a Laplacian matrix and calculating its eigenvector values, spectral methods can find each node's corresponding community accurately.

With in-depth analysis and research of complex network, the content information of complex networks shows its value and some community detection methods, which integrate the content information with network topology, have been developed. For instance, a subgraph overlapping clustering algorithm combining network structure and content information is proposed [9]. This method applies expectation-maximization (EM) algorithm to maximize likelihood function to generate stationary candidate subgraphs, and then uses k-means algorithm to cluster edges in order to obtain the overlapping community structure. A new generative probabilistic model is proposed which is learned by using a nested expectation-maximization algorithm and can describe the generalized communities [10]. In [11], a co-learning strategy is developed to jointly train the two parts (communities and semantics) in the model by combining a nested EM algorithm and belief propagation.

Recently, researchers have also realized that community detection should not only find communities, but also use rich verbal information in the text to give semantic description of communities. The description information reveals why some nodes gather into a community and helps people better understand the functions or implications of communities. For example, the approach in [12] using nonnegative matrix factorization integrates two tasks of community detection and user profiling into a unified model, and then achieves community profiling by a linear operator integrating the profiles of users. A joint community profiling and detection (CPD) model [13] is proposed which describes communities by published content and friendship links of users. In addition, the method SCI [14], which can detect and describe communities, has also been proposed. This method uses nonnegative matrix factorization to integrate topology and content information into a unified model, and achieves relatively high detection accuracy in comparison with other methods. More importantly, SCI can not only detect communities, but also analyzes the semantics of detected communities. In general, this type of method has more practical value than others without semantics.

However, the methods mentioned above mainly focus on how to effectively fuse topology structure and content information to improve the performance of community detection while do not further consider how to detect communities more robustly, especially when the node contents do not match well with network communities. Moreover, most of these methods can only interpret each community using a single topic, which is far from satisfactory in many real applications.

3 RSECD: The Network Model

Our proposed RSECD approach extends the previous SCI approach by introducing a transition probability matrix with a suitable prior to represent the hidden relationship between network communities and content clusters. In this section, firstly, we illustrate the difference between traditional generative model and our proposed new generative model; then we give some notations. Finally, we elaborate how to model RSECD.

3.1 Traditional Generative Model vs. New Generative Model

Most of community detection methods [9–14] follow traditional generative model which generally assumes that network topology and node contents share the same community structure (as shown in Fig. 1(a)). While in many real-world networks, network topology and node contents may implicate different community structures, so that we modify the traditional generative model and design a more reasonable generative model, as shown in Fig. 1(b). In this new model, node contents N implicates topic cluster T (not community structure C) and topic cluster T is generated by community structure C and transition probability matrix X together.

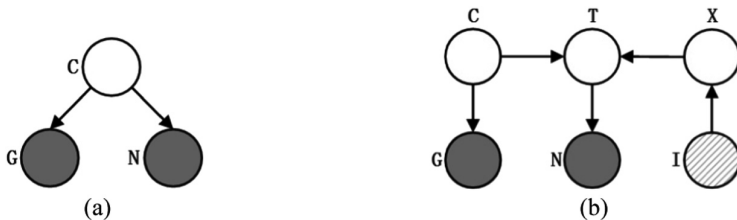


Fig. 1. A comparison of traditional generative model and our proposed new model. (a) is the traditional generative model where community structure C directly generates network topology G and node contents N. (b) is RSECD’s generative model where node contents N implicates topic cluster T (not community structure C) and topic cluster T is generated by community structure C and transition probability matrix X together. In addition, identity matrix I, as the transition matrix’s prior, plays a key guiding role in fusing these two types of information.

3.2 Notations

For an undirected network G with n nodes and e edges, we represent it by a binary-valued adjacency matrix $A \in \mathbb{R}^{n \times n}$. Each node i has its attributes S_i , which may be the

semantic information of the node. S_i is in the form of an m -dimensional binary-valued vector. All of S_i form an attribute matrix $S \in \mathbb{R}^{n \times m}$. The community detection task is: when A and S are observed, on topology, we need to find k different communities; on content cluster with semantics, we need to find k' different topics and infer the semantics for each community. Because all of the baseline algorithms assume that the number of communities is equal to that of topics, we still assume $k = k'$ in this paper. However, our RSECD algorithm can also apply equally to $k = k'$.

3.3 Modeling Network Topology

Our network topology model is based on the following intuitive properties: (1) if two nodes belong to the same community, they are more likely to be connected; (2) if two nodes have similar community memberships, they have a high probability to be linked. We define the propensity of node i belonging to community c as u_{ic} . Then we have a community membership of all nodes denoted as $U = (u_{ic})_{n \times k}$. Based on the first propensity, we can use $u_{ic}u_{jc}$ to represent the expected number of edges between nodes i and j in community c . Based on the second propensity, we can achieve that the expected number of edges between nodes i and j in the whole network is $\sum_{c=1}^k u_{ic}u_{jc}$. Considering all nodes, we have the following loss function:

$$\min_{U \geq 0} \|A - UU^T\|_F^2 \quad (1)$$

3.4 Modeling Node Attributes

We define the propensity of topic t having attribute q as c_{qt} and the propensity of node i belonging to topic t as v_{it} . Then we have an attribute membership of all topics denoted as $C = (c_{qt})_{m \times k}$ and a topic cluster membership of all nodes denoted as $V = (v_{it})_{n \times k}$. In addition, we define the propensity of a node i having attribute q as s_{iq} , which is an element of attribute matrix S . We suppose that if node i belongs to topic t , node i and topic t will have similar attributes information. It can be represented as $s_{iq} = \sum_{t=1}^k v_{it}c_{tq}$. Then we have the following loss function:

$$\min_{C \geq 0, V \geq 0} \|S - VC^T\|_F^2 \quad (2)$$

3.5 Modeling Transition Probabilities

Transition probability is an important concept of Markov chain and is defined as the probability of transferring from one state to another. We introduce transition probabilities to represent the relationship between network communities and topic clusters. Here the probability transferring from community c to topic t is defined as x_{ct} , the probability vector transferring from community c to any topic is defined as x_c (x_c satisfies a probability distribution) and the probability matrix transferring from any community to any topic is defined as X . Moreover, to effectively guide the fusion of

topology and content, we employ identity matrix I as the prior of X . Then we have the following loss function:

$$\min_{X \geq 0} \|UX - V\|_F^2 + \|X1_k^T - 1_k^T\|_F^2 + \|I - X\|_F^2 \tag{3}$$

where $1_k^T \in \mathbb{R}^{k \times 1}$ and all of its elements are 1.

3.6 The Unified Model

By combining the objective functions of the above formulas (including (1) to (3)), we obtain RSECD’s overall loss function:

$$\min_{\substack{U \geq 0, V \geq 0, \\ C \geq 0, X \geq 0}} L = \|A - UU^T\|_F^2 + \alpha \|S - VC^T\|_F^2 + \|UX - V\|_F^2 + \|X1_k^T - 1_k^T\|_F^2 + \|I - X\|_F^2 \tag{4}$$

where α is a balance parameter between network topology and node contents.

Our RSECD model can deal with the topology and content’s mismatch problem in networks well. To be specific, (1) when topology matches with content very well, the first two parts of unified model (network topology model and node attributes model) work so that topology and content can reinforce each other in order to find more exact community structure. (2) When only some parts of content match with network topology, RSECD can also extract useful material from content information to assist topology information in detecting more actual and accurate communities by the mapping and tractive function of transition matrix X . (3) When content does not match with topology at all, matrices U and V are almost orthogonal, thus matrix X is close to a random matrix and the final result is equal to that of using only topology information. In addition, the optimized X essentially represents the mapping relationship between communities and topics, so that we can also use X to explain the detected communities. So our RSECD is robust and strong-explanatory to community detection. We will further use extensive experiments (including a case study) to demonstrate these cases.

4 Optimization

Since the objective function in (4) is not convex, it is hard to obtain the global optimal solution. Fortunately, the local minima of (4) can be obtained using the Majorization-Minimization framework [16]. Here we describe an algorithm that iteratively updates U with V, C, X fixed, updates V with U, C, X fixed, updates C with U, V, X fixed, and updates X with U, V, C fixed, which guarantees that our objective does not increase and the parameters keep nonnegative (with any nonnegative initial seeds) after each iteration. The specific formulas are shown in the following subproblems.

4.1 U-Iteration

When updating U, we need to solve the following problem:

$$\min_{U \geq 0} L(U) = \|A - UU^T\|_F^2 + \|UX - V\|_F^2 \tag{5}$$

An arbitrary matrix M satisfies $\|M\|_F^2 = \text{tr}(MM^T)$, so we transform this problem as:

$$L(U) = \text{tr}(A^T A - A^T U U^T - U U^T A + U U^T U U^T) + \text{tr}(X^T U^T U X - X^T U^T V - V^T U X + V^T V) \tag{6}$$

We then take a derivative with respect to U and get the following formula:

$$\frac{\partial L(U)}{\partial U} = -2(A^T + A)U + 2(UX - V)X^T + 4UU^T U \tag{7}$$

In order to reduce computational cost, we use a multiplicative update algorithm based on the Oja’s iterative learning rule [15] to update U. We decompose (7) into two sets:

$$\nabla_U L(U) = \nabla_+ - \nabla_- \tag{8}$$

where ∇_+ (∇_-) is the sum of all positive (negative) components, then we have:

$$U_{\text{new}} = U_{\text{old}} \frac{\nabla_-}{\nabla_+} \tag{9}$$

In (7), the negative terms are $2A^T U$, $2AU$, $2VX^T$ and the positive terms are $2UXX^T$, $4UU^T U$. So we have the updating rule of U as:

$$u_{ij} \leftarrow u_{ij} \left(\frac{A^T U + AU + VX^T}{UXX^T + 2UU^T U} \right)_{ij} \tag{10}$$

4.2 V-Iteration and C-Iteration

When updating V, we need to solve the following problem:

$$\min_{V \geq 0} L(V) = \alpha \|S - VC^T\|_F^2 + \|UX - V\|_F^2 \tag{11}$$

In order to iterate V, we transform this problem into the following equation:

$$L(V) = \alpha \cdot \text{tr}(S^T S - S^T V C^T - C V^T S + C V^T V C^T) + \text{tr}(X^T U^T U X - X^T U^T V - V^T U X + V^T V) \tag{12}$$

We then take a derivative with respect to V and get the next formula:

$$\frac{\partial L(V)}{\partial V} = -2\alpha SC - 2UX + 2\alpha VC^T C + 2V \quad (13)$$

Similar to (10), we then obtain the updating rule of V as:

$$v_{ij} \leftarrow v_{ij} \left(\frac{\alpha SC + UX}{\alpha VC^T C + V} \right)_{ij} \quad (14)$$

When updating C , similar to the steps from (11) to (14), we obtain the updating rule of C as:

$$c_{ij} \leftarrow c_{ij} \left(\frac{S^T V}{C V^T V} \right)_{ij} \quad (15)$$

4.3 X-Iteration

When updating X , we need to solve the following problem:

$$\min_{X \geq 0} L(X) = \|UX - V\|_F^2 + \|I - X\|_F^2 + \|X1_k^T - 1_k^T\|_F^2 \quad (16)$$

To iterate X , we can transform this problem into the following equation:

$$\begin{aligned} L(X) = & \text{tr}(X^T U^T U X - X^T U^T V - V^T U X + V^T V) \\ & + \text{tr}(1_k X^T X 1_k^T - 1_k X^T 1_k^T - 1_k X 1_k^T + 1_k 1_k^T) + \text{tr}(I - X - X^T + X^T X) \end{aligned} \quad (17)$$

We then take a derivative with respect to X and get the next formula:

$$\frac{\partial L(X)}{\partial X} = -2U^T V - 2I - 2M + 2U^T U X + 2X M + 2X \quad (18)$$

where $M \in \mathbb{R}^{k \times k}$ and its elements are all 1. In (18), the negative terms are $2U^T V$, $2I$, $2M$ and positive terms are $2U^T U X$, $2X^T M$, $2X$. So we obtain the updating rule of X :

$$x_{ij} \leftarrow x_{ij} \left(\frac{U^T V + I + M}{U^T U X + X M + X} \right)_{ij} \quad (19)$$

5 Experiments

Here we first use artificial networks to analyze the influence of parameter α in the objective function and demonstrate that our approach can solve the mismatch problem well. We then compare our method with eight state-of-the-art algorithms on seven real datasets in terms of four well-known metrics. And finally, we discuss a case study analysis to show that our method has a strong explanatory capability to communities.

5.1 Artificial Networks

We use the Newman’s model [2] to generate artificial benchmark networks. Each network has 128 nodes which have been divided into 4 communities. Each node has z_{in} edges connecting to the nodes of the same community and z_{out} edges connecting to the nodes of different communities ($z_{\text{in}} + z_{\text{out}} = 16$). In addition, all nodes are partitioned into 4 clusters corresponding to 4 communities. To be specific, for each node in the s th cluster, we use a binomial distribution with mean $p_{\text{in}} = h_{\text{in}}/h$ to generate a h -dimensional binary vector as its $((s - 1) \times h + 1)$ -th to $(s \times h)$ -th attributes and use a binomial distribution with mean $p_{\text{out}} = h_{\text{out}}/(3 h)$ to generate its rest attributes. In our experiment, we set $h = 50$, $z_{\text{out}} = h_{\text{out}} = 8$ and use normalized mutual information (NMI) [19] as the metric. To simulate real-world networks’ mismatch problem, we use p_{mis} (ranging from 0 to 1) to reveal the mismatch rate between network topology and node contents. For example, if $p_{\text{mis}} = 0.8$, then in this network, there are 20% of nodes whose contents match with topology and 80% of nodes whose contents do not match with topology. In the first experiment, based on experience, we consider four choices for parameter α ($\alpha = 1$, $\|A\|_F^2$, $1/\|S\|_F^2$, or $\|A\|_F^2/\|S\|_F^2$) and respectively compute the average NMI values under them. The results are shown in Fig. 2(a), when p_{mis} is less than 0.6 (this corresponds to most cases in real-world networks), the result under $\alpha = \|A\|_F^2$ is greater than the others, so we conclude that choosing $\alpha = \|A\|_F^2$ as the default value may be better than the other three choices.

Next, to illustrate RSECD’s robustness, we compare three methods—Topo, SCI and RSECD. Topo is a variant of RSECD using topology information alone. SCI is a NMF-based method using topology and content information together but did not consider the mismatch problem [14]. As shown in Fig. 2(b), Topo keeps a stable detection accuracy no matter how p_{mis} changes because the topology information existing in the network is fixed. When p_{mis} is less than 0.3, as SCI combines topology and content information together, it has higher accuracy than Topo. However, because SCI fails to solve the mismatch problem, when p_{mis} is greater than 0.4, the performance of SCI gradually weakens and is worse than Topo. RSECD, as the extended work of SCI, has better performance than Topo and SCI when p_{mis} is less than 0.7. Moreover, when p_{mis} is larger than 0.7 (i.e., a high mismatch rate in the network), RSECD is just slightly worse than Topo but much better than SCI. In summary, the result demonstrates that: (1) when content match with topology well, RSECD can better combine topology and content to find communities; (2) when content does not match with topology, RSECD can also solve the mismatch problem well. Therefore, RSECD is robust.

Finally, because the cluster structure implicated by content information may be indistinct in the real-world networks, we design a third experiment. In this part, we set $p_{\text{mis}} = 0$ and relieve the constraint $h_{\text{out}} = 8$, making h_{out} vary from 0 to 12. The larger h_{out} is, the higher distinct degree is. The final result is shown in Fig. 2(c). As we can see, RSECD’s accuracy is almost always higher than that of SCI. Even though when the cluster structure is very indistinct, RSECD’s accuracy does not decline too much and is very close to that of Topo.

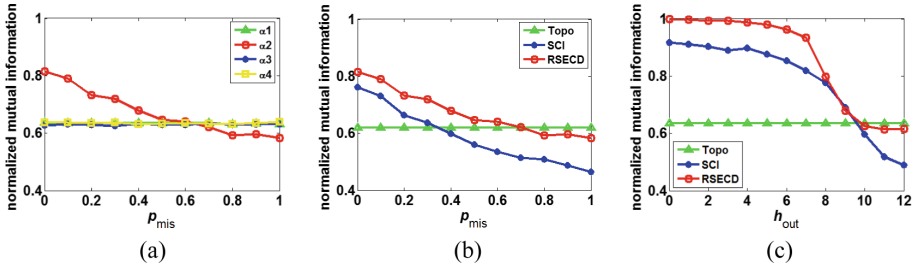


Fig. 2. Results on artificial networks. (a) is the NMI results under 4 different choices of parameter α . (b) is 3 different methods’ NMI results when the mismatch rate p_{mis} varies from 0 to 1. (c) is 3 different methods’ NMI results when h_{out} varies from 0 to 12 under $p_{\text{mis}} = 0$.

5.2 Real-World Networks

Datasets. We use 7 real networks [17, 18] with node attributes and ground-truth community labels. These datasets are often used in the field of community detection by researchers and their detailed information is shown in Table 1. In this table, the number of attributes represents the total number of attributes in the network.

Table 1. Datasets used.

Dataset	Communities	Nodes	Edges	Attributes	Ground truth
Facebook	14	226	3,417	131	✓
Cornell	5	877	1,608	1,703	✓
Texas	5	877	1,608	1,703	✓
Washington	5	877	1,608	1,703	✓
Wisconsin	5	877	1,608	1,703	✓
Citeseer	6	3,312	4,732	3,703	✓
Uai2010	19	3,363	45,006	4,972	✓

Metrics. To test RSECD’s performance, we conduct a quantitative analysis of the final detection results using two types of metrics (disjoint community metrics and overlapping community metrics). For disjoint community metrics, we choose accuracy (AC) [19] and normalized mutual information (NMI) [19]. AC is used to measure the percentage of correct labels obtained. In clustering applications, NMI is used to measure how similar two sets of clusters are. For overlapping community metrics, we choose F-score [20] and Jaccard similarity [20]. Both of them are common metrics which are used to quantify the performance in terms of the agreement between the ground-truth communities and the detected communities.

Baselines. To illustrate RSECD’s effectiveness, we choose three types of baseline algorithms including two topology-based methods (DCSBM [21] and BigCLAM [22]), one content-based method (AP [23]), and five methods using both topology and content (CESNA [24], DCM [25], PCL-DC [26], Block-LDA [27] and SCI [14]).

Setting. In the experiments, first for each network we uniformly set α to be $\|A\|_F^2$ based on previous parameter analysis. We then repeat RSECD algorithm 20 times with different random seeds. We obtain the result which corresponds to the smallest loss function value as the final result.

Table 2. Performance comparison of different methods using disjoint community metrics. Here “topo”, “cont”, “both” denote methods using topology, contents, and topology-and-contents.

Metrics (%)	Methods		Datasets					
	Type	Name	Cornell	Texas	Washington	Wisconsin	Citeseer	Uai2010
AC	topo	DCSBM	37.95	48.09	31.80	32.82	26.57	2.60
	both	PCL-DC	30.26	38.80	29.95	30.15	24.85	28.82
	both	Block-LDA	46.15	54.10	39.17	49.62	24.35	16.04
	both	SCI	36.92	49.73	46.09	46.42	29.53	29.51
	both	RSECD	53.85	61.50	58.70	69.43	48.67	47.21
NMI	topo	DCSBM	9.69	16.65	9.87	3.14	4.13	31.22
	cont	AP	25.27	31.02	31.79	32.48	13.28	41.60
	both	PCL-DC	7.23	10.37	5.66	5.01	2.99	26.92
	both	Block-LDA	6.81	4.21	3.69	10.09	2.42	5.70
	both	SCI	6.80	12.49	6.83	13.28	7.17	23.39
	both	RSECD	30.24	32.67	35.10	45.32	22.34	45.73

Table 3. Performance comparison of different methods using overlapping community metrics.

Metrics (%)	Methods		Datasets						
	Type	Name	Cornell	Texas	Washington	Wisconsin	Facebook	Citeseer	Uai2010
F-score	topo	DCSBM	34.08	36.14	32.83	29.47	44.92	26.83	30.12
	topo	BigCLAM	13.23	20.64	13.35	12.84	47.40	9.30	16.99
	cont	AP	21.10	23.59	24.11	20.53	23.60	12.92	13.23
	both	CESNA	23.48	23.54	21.91	23.17	52.51	3.38	32.32
	both	DCM	14.38	11.15	12.45	10.45	41.29	2.50	9.65
	both	PCL-DC	32.03	34.30	30.38	27.83	39.49	25.49	29.71
	both	Block-LDA	36.77	32.55	28.95	31.36	39.57	22.49	18.58
	both	SCI	26.94	30.99	28.06	27.06	24.94	26.18	29.66
	both	RSECD	53.26	44.89	47.44	53.54	52.73	45.77	43.86
Jaccard	topo	DCSBM	21.20	24.14	20.06	17.92	32.18	15.78	18.81
	topo	BigCLAM	7.18	12.18	7.25	7.01	34.25	5.01	9.87
	cont	AP	13.32	16.39	16.26	12.51	13.63	7.39	7.88
	both	CESNA	13.47	13.57	12.40	13.14	39.82	1.73	21.26
	both	DCM	7.95	6.03	6.72	5.54	33.60	1.27	5.77
	both	PCL-DC	19.02	21.56	18.99	16.27	26.99	14.75	19.17
	both	Block-LDA	24.29	22.51	18.20	20.31	26.61	12.80	11.08
	both	SCI	17.10	21.98	18.72	17.15	15.65	15.26	19.11
	both	RSECD	37.12	33.32	34.04	41.47	41.67	31.49	32.39

Results. We show the final results in Tables 2 and 3. It is worth noting that AP cannot compute accuracy (AC) value, and CESNA and DCM are only applicative to overlapping community metrics. In the tables, we use bold to mark the best results. Table 2 shows the comparison results in terms of AC and NMI. In AC, our method RSECD performs best among all the five methods. In NMI, RSECD still achieves the best results in comparison to the other methods. All the comparison results using different algorithms under overlapping community metrics are shown in Table 3. In these results, RSECD again has the best performance in comparison to the other tested approaches. In summary, the main reasons that our algorithm achieves such superior performance are as follows: (1) RSECD assumes that topology and content do not share the same community structure, so that those harmful content information will not interfere with topology information’s important role in community detection; (2) transition probability matrix, as a filter of content information, can retain beneficial content information which can assist topology information in detecting more actual, accurate communities and remove harmful content information which has wrong guidance in community detection. Therefore, RSECD can solve the mismatch problem well and the final performance results are relatively high and stable in any case.

Efficiency. As like standard nonnegative matrix factorization, the calculational complexity of RSECD is $O(T(n^2k + 2mnk + nk^2))$ where T is the number of iterations, n the number of nodes, k the number of communities ($k \ll n$) and m the number of attributes. By taking into account the sparsity of the adjacency matrix A and attribute matrix S , RSECD needs $O(T(ek + 2e'k + nk^2))$ time where e is the number of edges ($e \ll n$) and e' the number of nonzero elements in the attribute matrix S ($e' \ll m$). Thus, the computational complexity of RSECD is near linear with the number of nodes. We also report RSECD’s running time. It needs 2.893 s (here “s” denotes seconds), 8.9 s, 8.233 s, 10.952 s, 14.041 s, 6248.029 s and 5760.169 s, respectively, on the datasets Facebook, Cornell, Texas, Washington, Wisconsin, Citeseer and Uai2010.

5.3 A Case Study on Lastfm

We select LASTFM dataset¹, which comes from a musical social network, as our dataset for the case study analysis. This dataset contains 1,892 users and the total number of attributes in the network is 11,946. These attributes reveal users’ favorite songs or singers. LASTFM does not have the ground-truth of community labels. While, all the methods used in this work need the number of communities to be given. So, as did in [14], we use Louvain method [28] to set the number of communities in this network to 38. Two vivid examples to interpret the communities derived are shown in Figs. 3 and 4 in the form of word clouds. Word clouds can graphically show different attribute words’ importance degree in one community in order to explain the current community’s semantics. That is, in a word cloud, the size of a word is proportional to the probability that it belongs to this community.

The first example is the 30th community which contains two dominant topics, i.e., topics 1 and 32. Topic 1, as shown in Fig. 3(a), is highly related to electronic pop

¹ <http://ir.ii.uam.es/hetrec2011/datasets.html>.

music. The total of “electronic”, “electropop” and “electronica” has a high proportion in all attribute words and illustrates that the theme of topic 1 is pop electronic music. In addition, “australian”, “8-bit”, “synth pop”, “big beat” and “dark pop” are different styles of pop electronic music. On the other hand, topic 32, as shown in Fig. 3(b), mainly denotes synth pop music. Synth pop music originates from “new wave”, “post-punk” and is popular in “80 s”. “new romantic” is a synth pop song of Taylor Swift. “depeche mode” is a British band in style of alternative dance and synth pop. “electroclash” is another name of “tech pop” which contains the style of synth pop. “synth” and “synth pop” also appear here. It is worth noting that, these two topics which corresponds to electronic pop music of multiple styles and synth pop music, respectively, both belong to electronic pop music although being the different branches. Therefore, the 30th community will be a group of fans adoring electronic pop music mainly including synth pop music.



Fig. 3. Word clouds for the 30th community. (a) denotes topic 1 and (b) denotes topic 32, both of which are dominant topics of the 30th community.

Our second example is the 16th community which contains three dominant topics, i.e., topic 13, 24 and 36. They are shown in Fig. 4(a), (b) and (c), respectively. Similar to the previous analysis, we found out that topic 13 is related to opera music (for example, “diva”, “female vocalist” appear here); topic 24 is related to country music and pop music (for example, “country”, “pop” appear here); and topic 36 is related to dance music (for example, “dance”, “disco” appear here). Simultaneously, these three topics have the same theme, i.e., female singer. So, we can conclude that the 16th community’s dominant topic is female singers and the three topics (topic 13, 24, 36) in this community all have their own accurate semantics, respectively. Specifically, topic 13, 24, 36 respectively reflects opera music, country music and dance music.

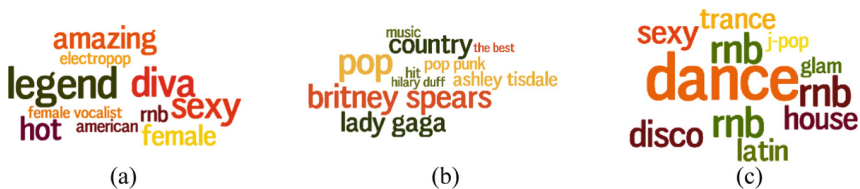


Fig. 4. Word clouds for the 16th community. This community contains three dominant topics, in which (a) denotes topic 13, (b) denotes topic 24 and (c) denotes topic 36.

6 Conclusion

In this paper, we proposed a new community detection method (RSECD) which is able to detect communities and in the same time analyze the semantics of founded communities. We introduced a nonnegative matrix factorization model to depict the relationships between nodes, topics and communities more accurately. A transition probability matrix with a suitable prior was also introduced to show their hidden relationships to improve the robustness of the new model, especially when node contents do not match well with network topology. Through artificial benchmark networks, we analyzed the influence of parameter α in the objective function and demonstrated RSECD's high level of robustness. On real-world networks, we showed that RSECD outperforms all of the baseline methods. Finally, the case study analysis on a musical social network showed how the semantic explanation of communities derived by RSECD works. This helps people to understand and interpret communities more precisely and in a human-readable form in many real applications.

Acknowledgment. This work was supported by the National Key R&D Program of China (2017YFC0820106), the Natural Science Foundation of China (61502334, 61772361, 61673293) and the Elite Scholar Program of Tianjin University (2017XRG-0016).

References

1. Fortunato, S., Hric, D.: Community detection in networks: a user guide. *Phys. Rep.* **659**, 1–44 (2016)
2. Girvan, M., Newman, M.: Community structure in social and biological networks. *Proc. Nat. Acad. Sci.* **99**(12), 7821–7826 (2002)
3. Jia, S., Gao, L., Gao, Y., et al.: Defining and identifying cograph communities in complex networks. *New J. Phys.* **17**(1), 013044 (2015)
4. Yang, L., Cao, X., He, D., et al.: Modularity based community detection with deep learning. In: *International Joint Conference on Artificial Intelligence (IJCAI)*, New York, USA, pp. 2252–2258 (2016)
5. Fanuel, M., Alaiz, C.M., Suykens, J.A., et al.: Magnetic eigenmaps for community detection in directed networks. *Phys. Rev. E* **95**(2), 022302 (2017)
6. Hao, F., Min, G., Pei, Z., et al.: K-clique community detection in social networks based on formal concept analysis. *IEEE Syst. J.* **11**(1), 250–259 (2017)
7. Whang, J.J., Gleich, D.F., Dhillon, I.S., et al.: Overlapping community detection using neighborhood-inflated seed expansion. *IEEE Trans. Knowl. Data Eng.* **28**(5), 1272–1284 (2016)
8. Jin, D., Wang, H., Dang, J., et al.: Detect overlapping communities via modeling and ranking node popularities. In: *30th AAAI Conference on Artificial Intelligence*, Phoenix, Arizona, USA, pp. 172–178 (2016)
9. Van Laarhoven, T., Marchiori, E.: Local network community detection with continuous optimization of conductance and weighted kernel k-means. *J. Mach. Learn. Res.* **17**(147), 1–28 (2016)
10. Jin, D., Wang, X., He, R., et al.: Robust detection of link communities in large social networks by exploiting link semantics. In: *32th AAAI Conference on Artificial Intelligence*, New Orleans, Louisiana, USA (2018)

11. He, D., Feng, Z., Jin, D., et al.: Joint identification of network communities and semantics via integrative modeling of network topologies and node contents. In: 31th AAAI Conference on Artificial Intelligence, San Francisco, California, USA (2017)
12. Akbari, M., Chua, T.S.: Leveraging behavioral factorization and prior knowledge for community discovery and profiling. In: Web Search and Data Mining (WSDM), UK, pp. 71–79 (2017)
13. Cai, H., Zheng, V.W., Zhu, F., et al.: From community detection to community profiling. *Proc. VLDB Endow.* **10**(7), 817–828 (2017)
14. Wang, X., Jin, D., Cao, X., et al.: Semantic community identification in large attribute networks. In: 30th AAAI Conference on Artificial Intelligence, Phoenix, Arizona, USA, pp. 265–271 (2016)
15. Oja, E.: Principal components, minor components, and linear neural networks. *Neural Netw.* **5**(6), 927–935 (1992)
16. Hunter, D.R., Lange, K.A.: A tutorial on mm algorithms. *Am. Stat.* **58**(1), 30–37 (2004)
17. Sen, P., Namata, G., Bilgic, M., et al.: Collective classification in network data. *AI Mag.* **29**(3), 93–106 (2008)
18. Leskovec, J.: Stanford Network Analysis Project (2016). <http://snap.stanford.edu>
19. Liu, H., Wu, Z., Li, X., et al.: Constrained nonnegative matrix factorization for image representation. *IEEE Trans. Softw. Eng.* **34**(7), 1299–1311 (2012)
20. Yang, J., Mcauley, J., Leskovec, J., et al.: Community detection in networks with node attributes. In: the IEEE International Conference on Data Mining series (ICDM), Dallas, Texas, USA, pp. 1151–1156 (2013)
21. Karrer, B., Newman, M.: Stochastic blockmodels and community structure in networks. *Phys. Rev. E* **83**(1), 016107 (2011)
22. Yang, J., Leskovec, J.: Overlapping community detection at scale: a nonnegative matrix factorization approach. In: Web Search and Data Mining (WSDM), Rome, Italy, pp. 587–596 (2013)
23. Frey, B.J., Dueck, D.: Clustering by passing messages between data points. *Science* **315**(5814), 972–976 (2007)
24. Yang, J., Mcauley, J., Leskovec, J., et al.: Community detection in networks with node attributes. In: the IEEE International Conference on Data Mining series (ICDM), Dallas, Texas, USA, pp. 1151–1156 (2013)
25. Pool, S., Bonchi, F., Van Leeuwen, M., et al.: Description-driven community detection. *ACM Trans. Intell. Syst. Technol.* **5**(2), 1–28 (2014)
26. Yang, T., Jin, R., Chi, Y., et al.: Combining link and content for community detection: a discriminative approach. In: 13th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD), Paris, France, pp. 927–936 (2009)
27. Balasubramanian, R., Cohen, W.W.: Block-LDA: jointly modeling entity-annotated text and entity-entity links. In: SIAM International Conference on Data Mining (SDM), Mesa, Arizona, USA, pp. 450–461 (2011)
28. Kido, G.S., Igawa, R.A., Barbon Jr, S.: Topic modeling based on louvain method in online social networks. In: Proceedings of XII Brazilian Symposium on Information Systems, Florianópolis, SC, pp. 353–360 (2016)



Dual Sum-Product Networks Autoencoding

Shengsheng Wang¹, Hang Zhang^{2(✉)}, Jiayun Liu¹,
and Qiang-yuan Yu¹

¹ College of Computer Science and Technology, Jilin University,
Changchun 130012, China

² College of Software, Jilin University, Changchun 130012, China
314362424@qq.com

Abstract. Sum-Product Networks (SPNs) are a new class of deep probabilistic model allowing tractable and exact inference. Recently SPNs have been successfully employed as autoencoder framework in Representation Learning. However, SPNs autoencoding mechanism ignores the model structural duality and train the models separately and independently. In this paper, we propose the Dual-SPNs autoencoding mechanism which design model structure as a dual close loop. This approach training the models simultaneously, and explicitly exploiting their structural duality correlation to guide the training process. As shown in extensive multilabel classification experiments, Dual-SPNs autoencoding mechanism prove highly competitive against the ones employing SPNs autoencoding mechanism and other stacked autoencoder architectures.

Keywords: Sum-Product Networks · Dual learning · Representation Learning
Multi-Label Classification

1 Introduction

The compromise between model expressiveness and tractability of model inference is a key issue of scientific computing [1]. Sum-Product Networks (SPNs) [2] were among the first learnable representations of these kind. SPNs are a deep generative probabilistic model that, by decomposing a probability distribution into sum and product nodes, allow tractable and exact computation of a series of probabilistic queries such as the conditionals, marginals and partition function. Based on those advantages, SPNs have very powerful performance in many AI tasks, such as natural language processing [3], computer vision [4].

So far, SPNs always as computational tractable model that have mainly been employed as “black box” inference model and distribution estimators. However, SPNs also have a powerful representation power in their inner nodes. Up to date, Vergari et al. [5] proposed a novel autoencoder framework base on SPNs and exploited their feature extraction performance for Representation Learning. They used SPNs encoder to encode sample into hidden representation. Moreover, they converted SPNs into Max-Product Networks (MPNs) decoder, provide a method to decode these representations back into the original input space. Representations learned by SPNs autoencoder are abundant hierarchical and probabilistic part-based features. But the

Sum-Product Networks Autoencoding mechanism has the following problems: (1) There is a strong structural duality between the SPNs encoder and MPNs decoder, this natural duality is largely ignored in the current method. (2) They train the SPNs encoder and MPNs decoder independently and separately, the feedback signals of the two models cannot be shared. Then a question arises: Can we exploit the duality between two models, so as to achieve better performance for both of them? In this work, we have a positive answer to this question.

Inspired by dual learning from natural language processing [6]. Dual learning trains two dual language translators (e.g., the primal task: English to French translator and the dual task: French to English translator) simultaneously by minimizing the reconstruction loss and used the policy gradient method of two dual translation tasks. The two dual translators represent closed loop. In this dual closed loop structure, the reconstruction loss measured over monolingual data (either English or French) would generate feedback signal to train a bilingual translator, even if without the involvement of a human labeler. Many AI tasks are emerged in dual forms, Machine Translation: translation from language A to language B vs. translation from language B to A; Image understanding: image captioning vs. image generation; Conversation: question answering vs. question generation. The dual property is not limited to dual tasks but also uses models with structural duality properties. The Autoencoder and conditional Generative Adversarial Networks (GANs) have the virtual duality. Their inner models have probabilistic correlation in dual form. DualGAN [7] has been successfully applied in image-to-image translation domain.

In this paper, we present a novel Dual-SPNs autoencoding mechanism. Firstly, we investigate the structural duality properties in SPNs autoencoder model and we proposed a Dual-SPNs autoencoding architecture: The Primal SPNs encode sample into hidden representation and Dual MPNs decode these representations back into the original input space. Dual-SPNs autoencoding mechanism is different from the SPNs autoencoding mechanism, because our architecture is a dual close loop. Then we propose training the Primal SPNs and Dual MPNs simultaneously, and explicitly leverages their structural duality correlation to regularize the training process. We demonstrate that Dual-SPNs autoencoding mechanism can improve the practical performances of both encoding and decoding processing in SPNs autoencoding mechanism. Additionally, Dual-SPNs autoencoding mechanism compared to traditional autoencoder architectures, they have the following advantages: (1) exactly answering a wider series of probabilistic queries in a tractable method; (2) a hierarchical, part-based and recursive definition in the model structural that allowed the extraction of rich and compositional representations well suited for image, texts and other natural data; (3) time and effort saved in hyperparameter tuning since both their weights and structure can be learned in a “cheap” way [8]. This makes Dual-SPNs autoencoding mechanism an excellent choice for Representation Learning. As a final contribution, the benefits of the resulting Dual-SPNs autoencoding routines are demonstrated by massive experiments on Multi-Label Classification tasks. Dual-SPNs autoencoding show surprisingly competitive performances when compared to those extracted from RBMs, probabilistic autoencoders and deep autoencoders tailored for label embeddings in all the learning scenarios evaluated.

2 Related Work

2.1 Sum-Product Networks

An SPN S over sets of random variables X is a new type of deep model consisting of a rooted DAG with interior nodes that are sum nodes and product nodes while the leaves nodes are tractable distributions. The edges emanating from each sum nodes to its children has a non-negative weight. All leaves of the SPN are distribution functions over some subset $Y \subseteq X$. When we know that a node N is a leaf, we also use the explicit symbol D . Inner nodes are either weighted sums or products, denoted as S and P , respectively, i.e., $S = \sum_{N \in \text{ch}(S)} \omega_{S,N} N$ and $P = \prod_{N \in \text{ch}(P)} N$, where $\text{ch}(n)$ denotes the children of N . The sum weights $\omega_{S,N}$ are assumed to be non-negative and normalized.

Formally, following [2] that we can define an SPN model as follows:

Definition 1 (Scope). The scope of an input distribution D is defined as the set of RVs Y for which D is defined: $\text{sc}(D) = Y$. The scope of an inner node N is recursively defined as $\text{sc}(N) = \bigcup_{N' \in \text{ch}(N)} \text{sc}(N')$.

To allow efficient inference, SPNs are required to fulfill two structure constraints, namely completeness and decomposability.

Definition 2 (Completeness). An SPN is complete if for each sum S it holds that $\text{sc}(N') = \text{sc}(N'')$, for each $N', N'' \in \text{ch}(S)$.

Definition 3 (Decomposability). An SPN is decomposable if all children of the same if it holds for each product P that $\text{sc}(N') \cap \text{sc}(N'') = \emptyset$, for each $N' \neq N'' \in \text{ch}(P)$.

An SPN valid only if all sum nodes are complete and all product nodes are decomposable, which guarantees that the value computed by the valid SPN for some evidence is proportional to the probability of that evidence.

While marginalization can be tackled in time linear in the network size, the problem of finding a Most Probable Explanation (MPE) is generally NP-hard in SPNs [2]. Given two sets of random variables $U, V \subset X, U \cup V = X, U \cap V = \emptyset$, inferring an MPE inference is defined as finding:

$$X_u = \arg \max_{u \sim U} p(u, v) \quad (1)$$

However, MPE can be solved exactly in Max-Product Networks. First one builds an MPN M from a SPN S by substituting each node $n \in S^\oplus$ by a max node $n \in M^{\max}$ computing $\max_{c \in \text{ch}(n)} \omega_{nc} M_c(x)$ and each leaf nodes distribution by a maximizing distribution (Fig. 1b). One then computes $M(x_{|O})$ -the MPE probability of the query $p(x_{|O})$ -by evaluating M bottom-up (Fig. 1c). Stage two consists of a top-down traversal of M . Starting from the root, one follows the maximal child branch for each max node and all child branches of a product node. Each partial input configuration determines a unique tree path. The MPE assignment x^* is obtained by collecting the MPE solutions (w.r.t. Q) of the leaves in the path (Fig. 1d). SPNs converted into MPNs provide a method (MPE inference) to decode these representations back into the original input space.

Parameters of an SPN can be learned generatively [2] or discriminatively [9] using Expectation Maximization or hard gradient descent. Building upon the currently most remarkable algorithm LearnSPN, a greedy top-down SPN learner introduced in [8]. LearnSPN proceeds by recursively decomposing a given data matrix along its rows (i.e. samples), generating sum nodes and estimating their weights, and its columns (i.e. RVs), generating product nodes.

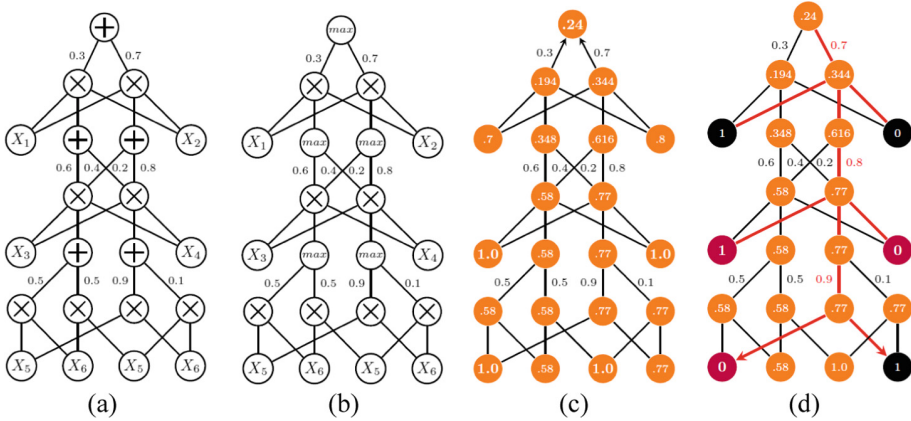


Fig. 1. A valid SPN S with leaves over univariate distributions labeled by their scopes (1a); the MPN M obtained from S (1b); and its bottom-up evaluation to solve $\arg \max_{q \sim Q} p(q, X_1 = 1, X_2 = 0, X_6 = 1)$ (2c) with $Q = \{X_3, X_4, X_5\}$. A tree path highlighted by MPE assignment in the top-down traversal of M (1d). Orange for inner activations. The assignment for random variables Q (resp. $E = \{X_1, X_2, X_6\}$) is the red (resp. black) leaves. (Color figure online)

2.2 Dual Learning

Dual learning as a new machine learning paradigm was proposed by He et al. [6]. In neural machine translation applications, the dual learning aimed to train two neural machine translators (French-English translator and English-French translators) simultaneously and the dual task is a close loop. In dual task, two agents play dual learning game is the essential idea of dual learning method (e.g. Agent A understands French only and Agent B understands English only). The two agents assessed the possibility of natural sentences being translated into the target language and the degree to which they could be reconstructed in accordance with the original text. The dual learning game is played alternatively on both sides, allowing translators to be trained from monolingual data only. Algorithms like policy gradient can be used to improve both primal and dual models according to feedback signals. Dual learning by learning from monolingual data (with 10% bilingual data for warm start), it achieves a comparable accuracy to neural machine translation trained from the full bilingual data for the French-to-English translation task.

Actually, the idea of “Dual Learning” is much more generally applicable even for tasks without physical duality. Now, we explore the method of dual learning and find that

many machine learning models are emerged in dual forms. The Autoencoder and conditional Generative Adversarial Networks (GANs) have the virtual duality. The Autoencoder have a Primal task: encoder f encode a function $y = f(x)$ (hidden representation) from raw data x . A Dual task: decoder g decode y and get a new data $x' = g(y)$. Feedback signal during the autoencoder loop: $R(x, f, g) = s(x, x')$. The GANs have a Primal task: the generator f generated fake data $y = f(x)$ from noise x . A dual task: the discriminator g discriminated the data from f whether natural or generated? The generator is receiving a feedback signal from the discriminator letting it know whether generated data is natural or not. Feedback signal: $R(x, f, g) = g(y) = g(f(x))$.

Probabilistic view of model structural duality, the primal-dual structure implies strong probabilistic connections between the two tasks. A primal task and its dual task, the primal task takes a sample from space X as input and maps to space Y , and the dual task takes a sample from space Y as input and maps to space X . Using the language of probability, the primal task learns a conditional distribution $P(y|x; \theta_{xy})$ parameterized by θ_{xy} and the dual task learns a conditional distribution $P(x|y; \theta_{yx})$ parameterized by θ_{yx} where $x \in X$ and $y \in Y$, the two dual tasks are jointly learned and their structural relationship is exploited to improve the learning effectiveness. The joint probability $P(x, y)$ can be computed in two equivalent ways:

$$P(x, y) = P(x)P(y|x) = P(y)P(x|y) \quad (2)$$

For any $x \in X, y \in Y$, ideally the conditional distributions of the primal and dual tasks should satisfy the following equality:

$$P(x)P(y|x; \theta_{xy}) = P(y)P(x|y; \theta_{yx}) \quad (3)$$

We use it as a regularization term to govern the training process. This can also be used to improve SPNs autoencoding mechanism, and perhaps even inference. Structural regularize to enhance SPNs autoencoding mechanism, additional criterion to improve inference.

3 Dual Sum-Product Networks Autoencoding

3.1 Network Configuration

In Fig. 2, we design model structure as a dual close loop and we use of a Primal SPN S to estimate $p(X)$ over some raw dataset $\{x^i \sim X\}_i$. We encoded each sample $x^i \sim X$ into a continuous vector representation e^i in a new d -dimensional space. To find an embedding function $f_S : X \rightarrow E_X$, we consider Primal SPNs. Given a Primal SPNs S and a set of nodes $N = \{n_j\}_{j=1}^d \subset S$, we construct our embedding as

$$e_j^i = S_{n_j}(x_{sc(n_j)}) = p_{w_{n_j}}(x_{sc(n_j)}) \quad (4)$$

Now we tackle the task to revert Primal-SPNs representations back into the input space. We used a Dual-MPNs to find a transformation $g : E_X \rightarrow X$ such that

$x^i \approx x^{i'} = g(f(x^i))$. We exploit a Dual-MPNs M and propose a procedure for g_M that mimic the decoding algorithm to compute the MPE inference in M . We notice that when a sample x^i is completely observed, then the computation of $M(x^i)$ activates only one maximal path in the network that can be traced back by a Viterbi-like procedure to a set of leaves whose scopes are a partition of X . We define the decoded function for a leaf n as the configuration over its scope that minimizes some distance D over the leaf activation value and its encoded representation:

$$x_{sc(n)}^{i'} = \arg \min_{u-sc(n)} D(\phi_n(u) || e_{M_n}^i) \tag{5}$$

We will employ an L_1 distance $|\phi(u) - e_{M_n}^i|$ in our experiments. It proved surprisingly effective in our experiments.

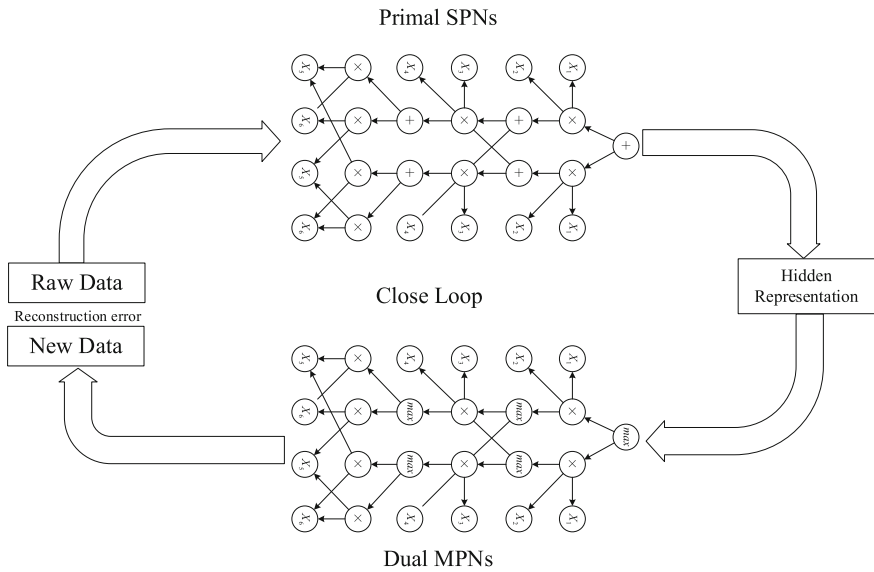


Fig. 2. The framework of Dual-SPNs autoencoding mechanism

3.2 Training Procedure

We learn the weights and structure in our Primal SPNs and Dual MPNs models by employing LearnSPN-b [10], as an improved version of LearnSPN algorithm. Firstly LearnSPN-b slices the data matrix into two parts, then checking for random variables independence and performing row clustering. In order to improve the greedy hierarchical clustering processes, it has proven to obtain simpler and deeper networks without limiting their performance and expressiveness. We define the same scopes for LearnSPN-b hyperparameters both when we learn our Primal SPNs and hence Dual MPNs.

For our framework of Dual-SPNs autoencoding mechanism, we describe the proposed algorithm in this subsection. Overall, the framework includes three components:

Primal SPNs, Dual MPNs and a regularization term that reflects the duality of Primal SPNs and Dual MPNs. Accordingly, the training objective of our framework includes three parts, which is described in Algorithm 1.

Algorithm 1 Dual-SPNs autoencoding training procedure

Input: Marginal distributions $\hat{P}(x_i)$ and $\hat{P}(y_i)$ for any $i \in [n]$; Lagrange parameters λ_{xy} and λ_{yx} ; optimizers Opt_1 and Opt_2 ;

Output: Primal SPNs f_s parameterized by θ_s ; Dual MPNs g_M parameterized by θ_M

Randomly initialize θ_s, θ_M

repeat

Get a minibatch of m pairs $\{x_j, y_j\}_{j=1}^m$;

Calculate the gradients as follows:

$$G_f = \nabla_{\theta_{xy}} \frac{1}{m} \sum_{j=1}^m [l_1(f_s(x_j; \theta_s), y_j) + \lambda_{xy} l_{duality}(x_j, y_j; \theta_s, \theta_M)];$$

$$G_g = \nabla_{\theta_{yx}} \frac{1}{m} \sum_{j=1}^m [l_2(g_s(y_j; \theta_M), x_j) + \lambda_{yx} l_{duality}(x_j, y_j; \theta_M, \theta_s)];$$

Update θ_s and θ_M

$\theta_s \leftarrow Opt_1(\theta_s, G_f)$, $\theta_M \leftarrow Opt_2(\theta_M, G_g)$.

until models converged

The Primal SPNs specific objective aims to minimize the loss function $l_1(f_s(x_j; \theta_s), y_j)$ and aimed to encode raw data to hidden representation. The Dual MPNs specific objective aims to minimize the loss function $l_2(g_s(y_j; \theta_M), x_j)$. The third objective is the regularization term which satisfies the probabilistic duality constrains as given in:

$$l_{duality} = (\log \hat{P}(x) + \log P(y|x; \theta_{xy}) - \log \hat{P}(y) - \log P(x|y; \theta_{yx}))^2. \quad (6)$$

In the algorithm, the choice of optimizers Opt_1 and Opt_2 from [11]. Then, we learn the models of Primal SPNs and Dual MPNs by minimizing the weighted combination between the original loss functions and the regularization term. While $l_{duality}$ can be regarded as a regularization term, it is data dependent. Every training sample contributes to the regularization term, and each model contributes to the regularization of the other model.

4 Experiment and Result Analysis

To evaluate Dual-SPNs autoencoding mechanism, we focus on Multilabel Classification (MLC) task. In MLC one is interested in predicting the target labels associated to a sample $x^i \sim X$ and represented as binary arrays: $y^i \sim Y$. In a simplest fully supervised

scenario, we trained a predictive model p , from the original feature space to the label one $X \xrightarrow{p} Y$. Instead, we can first encode both the input original feature X and/or the target label Y into different embedding spaces E_X, E_Y , and build a predictive model on top of them. In order to do so, we define different settings: we learn a predictive model on the input embeddings instead of the raw features ($E_X \rightarrow Y$); alternatively, one can first train a regressor on the original input X to predict label embeddings ($X \rightarrow E_Y$), then decoding such predictions back to the original label space; finally, the same regressor can be trained on the input embeddings instead ($E_X \rightarrow E_Y$) and its predictions decoded as above. we employ an L_2 -regularized logistic regressor (LR), (resp. a ridge regressor, RR) to predict each label in Y (resp. component in E_Y) independently. Therefore, the most natural baseline to measure the aforementioned representation meaningfulness is to employ the same L_2 -logistic regressor to the $X \xrightarrow{p} Y$ setting.

We now introduce other models as either encoders or encoder/decoders to plug into our settings. We employ a fully supervised method max-margin structured SVMs employing CRFs (CRF_{SSVM}) [12] in the $X \xrightarrow{p} Y$ as baseline. For the $E_X \rightarrow Y$ setting, we consider RBMs with 500, 1000, 5000 hidden units (h). A natural competitor for all settings are MADEs [13] because they are deep autoencoders which are also tractable probabilistic models. We employ MADEs comprising 3 layers and 500 and 1000 (resp. 200 and 500) hidden units per layer for the $E_X \rightarrow Y$ (resp. $X \rightarrow E_Y$) setting. Additionally, we add to the comparison MANIAC [14] a non-probabilistic autoencoder model tailored to MLC in our $X \rightarrow E_Y$ setting. Lastly, we use traditional SPNs autoencoding embeddings in $E_X \rightarrow Y, X \rightarrow E_Y$ and $E_X \rightarrow E_Y$ setting.

We measure the JACCARD, HAMMING and EXACT MATCH scores, as metrics highly employed in the MLC. For all experiments we use 11 standard Multilabel Classification data sets in Yahoo!, web page collections described in [15] are very popular for MLC. The Yahoo! data sets consist of 11 subdirectory data sets: Arts, Business, Computer, Education, Entertainment, Health, Recreation, Reference, Science, Social&Science, Society&Culture. To fairly compare all the algorithms in our experiments, we employ the binarized versions of all datasets and divided in 5 folds already processed by Label-Attribute Interdependence Maximization (LAIM) [16]. LAIM is a discretization method for multi-label data.

In Table 1, we report the average scores over all datasets in the form of the average relative improvement w.r.t. the $X \xrightarrow{LR} Y$ baseline. The best models for each setting and score are in 5 folds, the higher their improvement, the better. We also report the performance of a fully supervised method CRF_{SSVM} for MLC. For all settings and measures the Primal-SPNs and Dual-MPNs embeddings prove to be highly competitive against all other models.

In summary, Primal-SPNs and Dual-MPNs embeddings proved to be highly competitive and even superior to all other models in the three settings and for all the scores. Even the fully supervised CRF_{SSVM} performance are comparable to the best Primal-SPNs/Dual-MPNs JACCARD (resp. HAMMING) score in method $E_X \rightarrow Y$ (resp. $X \rightarrow E_Y$) setting.

To better understand the effects of applying the probabilistic structural duality constraint as the regularization, the $l_{duality}$ plays an important role in Dual-SPNs

autoencoding mechanism that can get a better JACCARD, HAMMING and EXACT MATCH scores in setting $E_X \rightarrow E_Y$) than traditional SPNs autoencoding mechanism. Representations from Dual-SPNs autoencoding mechanism even with smaller embeddings than RBMs and MADEs, yield the largest improvements. In setting $X \rightarrow E_Y$, disentangling the relationships among the Y gives all models a performance boost. This is not the case for MADEs on some datasets, probably due to their reconstruction power being traded off to their generalization power as generative models. Dual-MPNs, on the other hand, consistently exploit the label representation space and do not provide overfitted reconstructions.

All in all, with these Multilabel Classification tasks we gathered empirical confirmation of the meaningfulness and practical usefulness of Primal-SPNs and Dual-MPNs embeddings. In setting $E_X \rightarrow E_Y$, the reported large improvements over the three scores cannot be due to Primal-SPNs/Dual-MPNs larger embedding sizes. In fact, their sizes are always comparable or smaller than RBM, MADE, ones since the latter max capacities have been chosen after Primal-SPNs/Dual-MPNs have been learned. Indeed, MADE log-likelihoods have proven to be higher than SPN ones on many datasets and comparable on the rest. We argue that the reason behind these results lies in the hierarchical part-based representations Dual-SPNs provide. Each embedding component is responsible for capturing only the significant feature portions according to its corresponding node scope.

Table 1. Average relative test set (percentages) improvement w.r.t. the LR baseline on best results in 5 bold for Multi-Label Classification.

E_X	E_Y	Predictor	Decoder	JAC	HAM	EXA
X	Y	LR	-	0.00	0.00	0.00
X	Y	CRF _{SSVM}	-	+15.72	+9.26	+102.25
RBM _{$h=500$}	Y	LR	-	-1.12	-2.14	-14.24
RBM _{$h=1000$}	Y	LR	-	+0.86	-0.86	-7.35
RBM _{$h=5000$}	Y	LR	-	+1.45	+0.26	-1.52
MADE _{$h=500$}	Y	LR	-	+1.15	+0.01	-7.04
MADE _{$h=1000$}	Y	LR	-	+2.25	+0.45	+2.88
SPN	Y	LR	-	+3.62	+0.50	+17.32
Primal-SPN	Y	LR	-	+5.41	+0.60	+23.41
X	MADE _{$h=200$}	RR	MADE	-30.14	+7.12	-29.60
X	MADE _{$h=500$}	RR	MADE	-30.25	+7.22	-28.62
X	MANIAC	RR	MABIAC	+5.55	+5.23	+95.70
X	MPN	RR	MPN	+11.62	+9.65	+96.30
X	Dual-MPN	RR	Dual-MPN	+16.18	+8.14	+96.77
MADE _{$h=500$}	MADE _{$h=200$}	RR	MADE	-28.17	+7.22	-28.04
MADE _{$h=500$}	MADE _{$h=500$}	RR	MADE	-27.45	+6.97	-27.15
MADE _{$h=1000$}	MADE _{$h=200$}	RR	MADE	-27.84	+6.98	-19.03
MADE _{$h=1000$}	MADE _{$h=500$}	RR	MADE	-27.25	+6.94	-25.14
SPN	MPN	RR	MPN	+14.57	+8.28	+106.64
Primal-SPN	Dual-MPN	RR	Dual-MPN	+16.02	+9.97	+110.75

5 Conclusion

In this work we investigated Dual-SPNs autoencoding mechanism under a Representation Learning lens, we propose the Dual-SPNs autoencoding mechanism which design model structure as a dual close loop. Additionally, we propose training the Primal SPNs and Dual MPNs simultaneously, and explicitly leverages their probabilistic correlation to regularize the training process. Experiments on Multilabel Classification tasks demonstrated that the resulting framework of Dual-SPNs autoencoding indeed produces meaningful features and is competitive to the SPNs autoencoding mechanism and other stacked autoencoder architectures.

There are multiple directions to explore in the future. First, combine Dual-SPNs autoencoding with dual inference so as to leverage structural duality to enhance both the training and inference procedures. Second, we will enrich theoretical study to better understand Dual-SPNs autoencoding mechanism.

Acknowledgments. This work is supported by the National Natural Science Foundation of China (61472161), Science & Technology Development Project of Jilin Province (20180101334JC).






References

1. Choi, A., Darwiche, A.: On relaxing determinism in arithmetic circuits. In: Thirty-Fourth International Conference on Machine Learning, pp. 825–833. ACM, Sydney (2017)
2. Poon, H., Domingos, P.: Sum-product networks: a new deep architecture. In: The 27th Conference on Uncertainty in Artificial Intelligence, pp. 337–346. AUAI, Barcelona (2011)
3. Cheng, W., Kok, S., Pham, H.: Language modeling with sum-product networks. In: Fifteenth Annual Conference of the International Speech Communication Association, pp. 2098–2102. ISSN, Singapore (2014)
4. Amer, M.R., Todorovic, S.: Sum product networks for activity recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **38**(4), 800–813 (2016)
5. Vergari, A., Peharz, R.: Sum-product autoencoding: encoding and decoding representations using sum-product networks. In: The Thirty-Second Conference on Artificial Intelligence. AAAI, New Orleans (2018)
6. He, D., Xia, Y., Qin, T., et al.: Dual learning for machine translation. In: The Thirtieth Annual Conference on Neural Information Processing Systems, pp. 820–828. MIT Press, Barcelona (2016)
7. Yi, Z., Zhang, H., Tan, P., et al.: DualGAN: unsupervised dual learning for image-to-image translation. In: International Conference on Computer Vision, pp. 2868–2876. IEEE, Venice (2017)
8. Gens, R., Pedro, D.: Learning the structure of sum-product networks. In: 30th International Conference on Machine Learning, pp. 873–880. ACM, Atlanta (2013)
9. Gens, R., Domingos, P.: Discriminative learning of sum-product networks, In: 26th Advances in Neural Information Processing Systems, pp. 3239–3247. MIT Press, Lake Tahoe (2012)
10. Vergari, A., Di Mauro, N., Esposito, F.: Simplifying, regularizing and strengthening sum-product network structure learning. In: Appice, A., Rodrigues, P.P., Santos Costa, V., Gama, J., Jorge, A., Soares, C. (eds.) ECML PKDD 2015. LNCS, vol. 9285, pp. 343–358. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-23525-7_21

11. Kingma, D., Adam, J.B.: A method for stochastic optimization. *Comput. Sci.* (2014)
12. Finley, T., Joachims, T.: Training structural SVMs when exact inference is intractable. In: *International Conference on Machine Learning*, pp. 304–311. ACM, Helsinki (2008)
13. Germain, M., Gregor, K., Murray, I., et al.: MADE: masked autoencoder for distribution estimation. In: *The 32nd International Conference on Machine Learning*, pp. 881–889. ACM, Lille (2015)
14. Wicker, J., Tyukin, A., Kramer, S.: A nonlinear label compression and transformation method for multi-label classification using autoencoders. In: Bailey, J., Khan, L., Washio, T., Dobbie, G., Huang, J.Z., Wang, R. (eds.) *PAKDD 2016. LNCS*, vol. 9651, pp. 328–340. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-31753-3_27
15. Ueda, N., Saito, K.: Single-shot detection of multiple categories of text using parametric mixture models. In: *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 626–631. ACM, Alberta (2001)
16. Cano, A., Luna, J.M., Gibaja, E.L.: LAIM discretization for multi-label data. *Inf. Sci.* **330**, 370–384 (2016)



Recognizing Diseases from Physiological Time Series Data Using Probabilistic Model

Danni Wang¹ , Li Liu¹ , Guoxin Su² , Yande Li³ , and Aamir Khan¹ 

¹ School of Big Data and Software Engineering, Chongqing University,
Chongqing 400044, People's Republic of China
{wdn,dcsliuli}@cqu.edu.cn, 141996aamirkhan@gmail.com

² School of Computing and Information Technology, University of Wollongong,
Wollongong, NSW 2522, Australia
guoxin@uow.edu.au

³ College of Information Science and Engineering, Lanzhou University,
Lanzhou 730000, People's Republic of China
liy2016@lzu.edu.cn

Abstract. Modern clinical databases collect a large amount of time series data of vital signs. In this work, we first extract the general representative signal patterns from physiological signals, such as blood pressure, respiration rate and heart rate, referred to as atomic patterns. By assuming the same disease may share the same styles of atomic patterns and their temporal dependencies, we present a probabilistic framework to recognize diseases from physiological data in the presence of uncertainty. To handle the temporal relationships among atomic patterns, Allen's interval relations and latent variables originated from Chinese restaurant process are utilized to characterize the unique sets of interval configurations of a disease. We evaluate the proposed framework using MIMIC-III database, and the experimental results show that our approach outperforms other competitive models.

Keywords: Disease pattern recognition · Physiological signals
Atomic pattern · Temporal relationship

1 Introduction

Nowadays, patient monitors are widely used to capture patients' physiological signals and vital signs in intensive care unit (ICU). With the increase of patients' physiological data collected in hospitals, it is possible to find the temporal patterns of physiological time series for different diseases, and thus understand the correlations between physiological data and medical conditions. In the last decade, the interest has mainly been focused on simple signal findings. Complex temporal relationships among multiple physiological signals, such as heart rate, respiration rate and blood pressure, have been recently applied to detect

diseases. For example, *if there is a rise of the blood pressure and an overlapping steady state of heart rate, followed by a decrease of respiration rate, does an individual suffer a heart failure?* Similar to complex activity recognition [9], where many motion sensor data are collected to detect individuals' activities, we can recognize certain diseases of different individuals by leveraging their corresponding physiological data.

In this paper, we first explore the frequent patterns appeared in a certain disease, referred to as atomic patterns, by clustering raw physiological data points from a long-term physiological signal. As shown in Fig. 1, seven most frequently appeared atomic patterns are extracted from each of the five physiological signals in respiratory failure disease. A series of physiological signals is a collection of consecutive atomic patterns. These signals may include various kinds of physiological temporally data collected from patients. A disease can be represented as a set of atomic pattern combinations and their temporal relations.

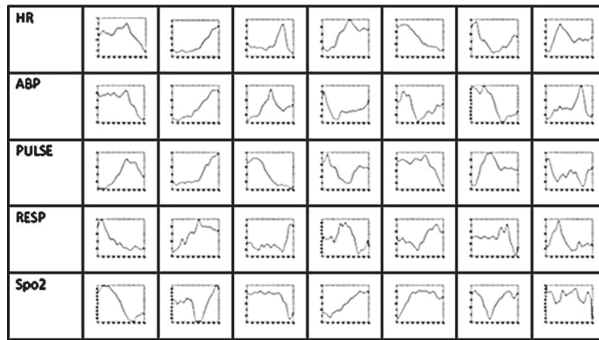


Fig. 1. Examples of the atomic patterns extracted from the five physiological signals (HR - heart rate; ABP - arterial blood pressure; PULSE - pulse; RESP - respiration rate; SpO2 - oxygen saturation) most frequently appeared in respiratory failure.

A disease recognition model should not only characterize the temporal dependencies between atomic patterns among physiological signals, but also represent the uncertainties corresponding to individual atomic patterns and their temporal dependencies. It is known that different individuals who suffer the same disease may possess a unique set of physiological signals, which are obviously different from the others. This kind of variability of diseases often shows themselves by the types of underlying atomic patterns and the temporal dependencies that exists between them.

To address these issues in disease recognition, our model specifies a joint probability distribution over atomic patterns and their interval relations. Particularly, we introduce a latent variable generated by the Chinese restaurant process [15], which enables our model to handle infinite and repetitive occurrences of atomic patterns. Overall, our generative model is more flexible than

discriminative models (e.g. KNN) in expressing rich interval relations in the complex learning task of disease recognition from physiological signals.

2 Related Work

Some classification methods which are not involving temporal relationships have been used to recognise diseases. In [14], Kathija et al. applied Naive Bayes and SVM to classify breast cancer as malignant or benign. In [13], Nikovsk et al. achieved medical disease recognition using Bayesian Networks with data which are not completely correct. In [12], Ni et al. proposed Cross-network Clustering and Cluster Ranking (*CCCR*) to diagnose diseases. In [3], Beumer et al. presented an overview of qualitative probabilistic networks in the context of skin diseases with children. Fatima et al. [4] compared the performances of various algorithms for diagnosing various diseases. In recent years, medical time-series physiological data has been researched via plenty of temporal relationships mining methods. Sacchi et al. achieved rule mining in biomedical data using a knowledge-based model [16]. In [7], a change detection based multivariate relation rule mining algorithm is presented for mining rules in complex dataset. Mudlikhah et al. presented an disease rule mining method combining fuzzy inference and subtractive clustering to mine rules from certain disease [11]. Banaee et al. presented a fully data-driven approach which is based on Allen's interval relations to extract and represent temporal relation of atomic patterns in clinical data streams and make a textual output [2]. The knowledge-based models are restricted by the knowledge of domain experts. The methods above are all less expressive to depict the uncertainties related to the temporal relationships. The interval temporal Bayesian network is an effective interval based graphic model combining probability description of Bayesian network and Allen's relations to recognize complex activities [17]. However, it cannot handle repetitive atomic patterns within one record of a disease.

3 Datasets

The data we used in this paper are selected from MIMIC-III waveform database. The MIMIC-III Waveform Database contains records of multiple time-series vital signs and physiological signals obtained from bedside monitors in intensive care units (ICUs) [8]. Each record belongs to one subject. The clinical information of each subject can be found in the related clinical database, including the disease that the subject is diagnosed as. In our experiment, three physiological signals were chosen to be analyzed, i.e., heart rate (HR), arterial blood pressure (ABP) and respiration rate (RESP), as shown in Fig. 2.

We chose records that contain minute-by-minute numeric time series of variant physiological signals. In addition, we chose the records of patients who suffered one of three diseases, which are heart failure, chronic respiratory failure and sepsis. Precisely, five waveform records of five patients who suffered heart

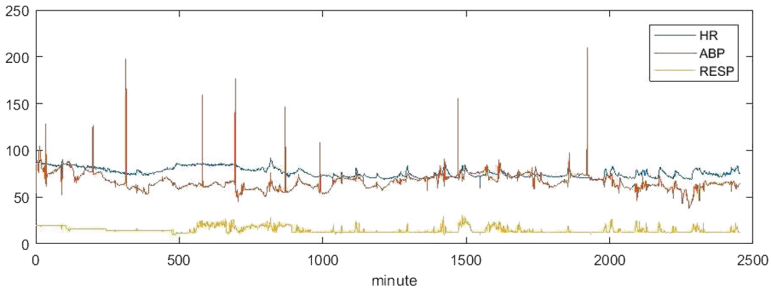


Fig. 2. Example of the raw sensor data from heart rate, blood pressure and respiration rate, respectively.

failure and four waveform records of four patients who suffered sepsis and respiratory failure were selected individually. The detailed information of the selected patients and their waveform records are listed in Table 1. To minimize the impact of other diseases, we chose records of the top three diseases the patients suffered. We faced some challenges using clinical records in the MIMIC-III database. Some of the raw data is incomplete, abnormal and sparse. To deal with the challenges [10], we preprocessed the records we chose before analyzing. The process includes discarding abnormal data points and smoothing time series. Another challenge we faced while collecting data was that only a small amount of waveform records had been matched to the MIMIC-III Clinical Database. This challenge resulted in small number of records used in our experiment.

Table 1. Our selected records.

Diseases	No. of subjects (records)	Average length of records
Heart failure	5 (5)	5929 min
Sepsis	4 (4)	2930 min
Resp failure	5 (5)	2915 min

4 Pattern Abstraction

One of the significant steps is pattern abstraction. Before pattern abstraction, data preprocessing is applied. We first discard the data with unreliable values, and then we applied Local Weighted Regression (LOESS) as a smoothing function to reduce noise among the data. Figure 3 shows the raw data and the data after *LOESS* processing. It can be seen that the sharps of the data are smoothed after *LOESS*.

In order to relate each time series to a corresponding atomic pattern sequence [5], segmentation method is being applied. Sliding Window is used during this process. We set the length of segments to 120, to achieve this, the

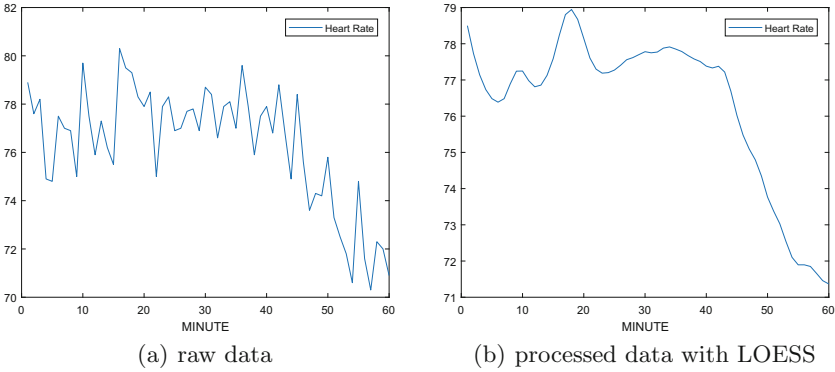


Fig. 3. Raw data and data processed with LOESS.

width of sliding window is set to 120. The overlap length of each two adjacent segments is set to 60, which is of half length of a segment. Assume we have time series data $T = \{t_1, t_2 \dots t_n\}$, after applying the segmentation steps above, discrete sequence of segments $P = \{p_1, p_2 \dots p_m\}$ is obtained, where generally $n \gg m$. Segmentation is followed by clustering. Segmented sequence of each signal is clustered for proper number of cluster centers which are treated as atomic patterns [6].

We merge segments of the same physiological signals from all the selected patients together, and then cluster the segments. K-means is used as clustering method. The centroid pattern clustered from the segments are atomic patterns. This is the first pattern abstraction process. Figure 4 shows all the centroid pattern clustered from the three kinds of time-series sensor data. After applying K-means, each segment is assigned with its most similar atomic pattern. After this step of clustering, we extract some features directly from the segments. The features include absolute maximum, absolute minimum, the number of local maximum, the number of local minimum, mean value, standard deviation, median of each segment. We form these values of features of each segments into new segments, each of which contains eight data points. Then we apply the clustering to the new segments to get the final sequences of atomic patterns, which is the final pattern abstraction process.

Figure 5 shows the complete procedure of our disease recognition approach.

5 Model Description

Interval Relation Description. The model we used in this paper is Probabilistic Interval-Based Bayesian Model with Allen’s relations. Supposing we have a dataset D of K records over F diseases, each record consists of atomic pattern interval sequences of several physiological measurements in the order of start-time, i.e. $\langle S_1, S_2, S_3, \dots, S_m \rangle$. Within a sequence, temporal relation between any two neighbouring atomic patterns can be before, meets, overlaps, starts, contains

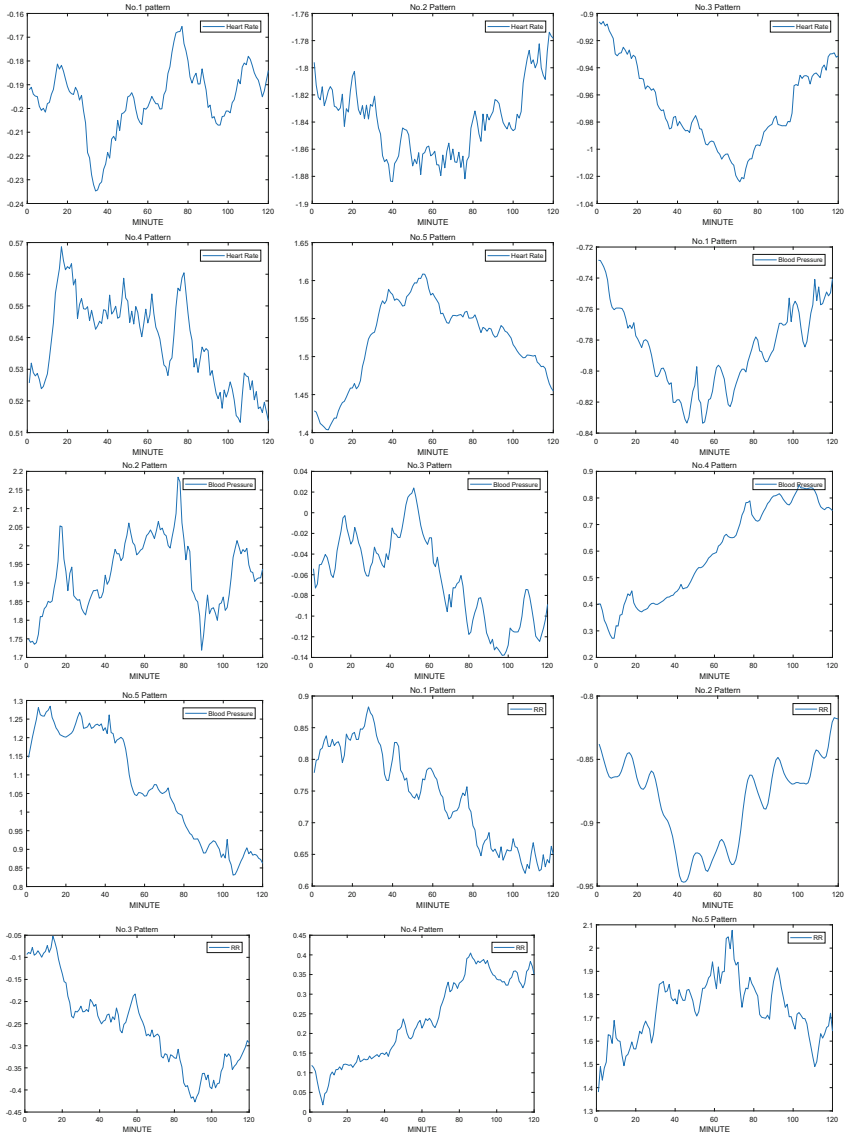


Fig. 4. Five atomic patterns of three physiological signals (heart rate, blood pressure and respiratory rate).

and finished-by. As these temporal relations are all included in Allen’s relations, Allen’s relations are chosen to describe the temporal relations between patterns in this paper. Between two intervals,thirteen possible temporal relations are formulated by Allen. They are *before*, *meets*, *overlaps*, *starts*, *contains*, *finished-by*, *equals*, *after*, *met-by*, *overlap-by*, *started-by*, *during* and *finishes*, denoted as

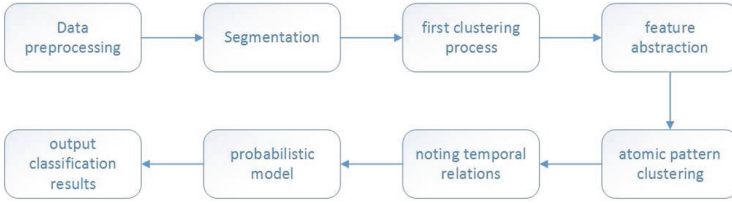


Fig. 5. Block diagram of disease recognition system.

$\{b, m, o, s, c, f, b^-, m^-, o^-, s^-, c^-, f^-, \equiv\}$ [1]. In this work we describe the interval relations using the non-negative Allen’s relations $\{b, m, o, s, c, f, \equiv\}$. Temporal relations between atomic patterns within a disease can be represented by an interval network. In the network, an interval is represented by a node, the Allen’s temporal relationship between two involved intervals is represented by a directed link. Only one temporal relation is associated to one such link, which is in $\{b, m, o, s, c, f, \equiv\}$. A directed cyclic graph is resulted. Within the graph, the temporal relations corresponding to links shall be in consistency. For example, p_1 starts p_2 and p_2 meets p_3 , the temporal relation on the link between p_1 and p_3 is *before*. The necessary and sufficient condition for an interval network is that the temporal relations on every pattern triangle meet transitivity properties.

One network can only represent one appearance of a disease. A disease can be shown as several appearances which can be represented by the probabilistic interval-based networks where atomic patterns and the temporal relations are formed differently. We want to build a model which can automatically construct the interval-based networks to characterize a disease.

In a dataset D , each record is associated with a set of Q atomic patterns, $P = \{P_1, P_2, P_3, \dots, P_Q\}$ and the seven non-negative relations $R = \{b, m, o, s, c, f, \equiv\}$. For each disease $f(1 \leq f \leq F)$, denote D_f which belongs to D the corresponding subset of K_f records. Each record $f \in D_f$ is an example of the f -th disease. Denote the quantity of intervals in the record f as $|f|$. Denote the network which describes f as $G_f = (V_f, E_f)$. Within G_f , E_f and V_f are a set of edges and nodes. In the network, each node is assigned with an atomic pattern from P , each edge is assigned with a relation from R . In our model, edges are generated only between two adjacent nodes. Therefore, a network G_f consists of $|f|$ nodes $v_{c,1}, v_{c,2}, \dots, v_{c,f}$ and $|f| - 1$ edges $e_{f,1,2}, e_{f,2,3}, \dots, e_{f,|f|-1,|f|}$.

Temporal Relations on Edges. Each node in the network is assigned with an atomic pattern. $Multinomial(\delta_{i,j})$ is being applied to decide the relation on edge $e_{f,m-1,m}$ from $R \{b, m, o, s, c, f, \equiv\}$, noting that subscript letter m represents the m -th node in network f , subscript letter i and j represent the i -th and the j -th atomic patterns in Q that are assigned to the $(m - 1)$ -th and m -th node in network f correspondingly. Since R includes seven temporal relations, each pair of atomic patterns (P_i, P_j) is set with a multinomial distribution parameter vector $\delta_{i,j}$ with seven dimensions. It can be known that relation $r_{f,i,j}$ on edge

$e_{f,m-1,m}$ only depends on two atomic patterns, P_i and P_j . The parameters for dataset D_f are learned using Maximum Likelihood Estimation (*MLE*). According to *MLE*, the likelihood of δ_{P_i,P_j} given dataset D_f is

$$L(\delta_{P_i,P_j}; D_f) = \prod_f P(r_{f,i,j} | P_i, P_j, \delta_{i,j}) = \prod_f \delta_{i,j,r}^{nr_{i,j,r}} \tag{1}$$

$\delta_{i,j} = \{\delta_{i,j,1}, \delta_{i,j,2}, \delta_{i,j,3}, \delta_{i,j,4}, \delta_{i,j,5}, \delta_{i,j,6}, \delta_{i,j,7}\}$. $nr_{i,j,r}$ is the sum of times when relation r appears between P_i and P_j . According to *MLE*,

$$\hat{\delta}_{i,j,r} = \frac{nr_{i,j,r}}{\sum_{k=1}^7 nr_{i,j,k}} \tag{2}$$

Node Generation. We used latent tables extracted from *Latent Dirichlet Allocation* and *Chinese Restaurant Process*. Assume the number of tables in a restaurant is limitless. Each node is assigned to a table. Each table contains Q atomic patterns with certain probabilities. Each atomic pattern has different probabilities on different tables. We assume on tables from the same set, atomic patterns relating to the same disease are more likely to be served. Therefore, nodes from the records of the same disease tend to be assigned to a group of tables where the atomic patterns relating to the disease are more likely to be served. We assume the first node is assigned to the first table. The process of the m -th node choosing a table shows below:

$$\begin{cases} \frac{nt_j}{n+\gamma-1} & (\text{if choose an occupied table}) \\ \frac{\gamma}{n+\gamma+1} & (\text{if choose an empty table}) \end{cases} \tag{3}$$

nt_j is the number of previous $n - 1$ nodes assigned to the table t_j , and γ is a tuning parameter which is positive. The node assignments to unoccupied tables are exchangeable and share the same probability in the distribution. According to Dirichlet Process, we set every table with different probabilities over those Q atomic patterns. Now, with an exclusive set of tables with the distributions over Q atomic patterns, a disease is identified. Each node $v_{f,m}$ choose a table following *CRP*(γ) as previously stated. When a node is assigned to a table t_j ($j = 1, 2, \dots$), an atomic pattern is chosen following the multinomial distribution *Multinomial*(Θ_j) to assign to the node. Θ_j is obtained from the a priori distribution *Dirichlet*(α), of which α is a hyperparameter. We set the maximum number of tables as z . Give each table a Dirichlet Distribution parameter vector α_g ($1 \leq g \leq z$). Therefore, *Dirichlet*(α) = $\{\text{Dirichlet}(\alpha_1), \text{Dirichlet}(\alpha_g), \dots, \text{Dirichlet}(\alpha_z)\}$.

To learn the latent variables and estimate parameters $\Theta_1, \Theta_2, \dots, \Theta_z$, we choose approximate inference, using Gibbs sampling as the method. After calculating the probability of assigning each node of the graph to each table, and sampling with the available tables, the distribution of $\Theta_{j,q}$ ($1 \leq j \leq z$) is estimated as

$$\Theta_{j,q} = \frac{na_{j,q}}{\sum_{q=1}^Q na_{j,q} + \alpha Q} \tag{4}$$

6 Evaluations

Experimental Setup. The experiments are implemented on one computer. The version of the CPU used in the experiments is Intel i5-4590@3.30 GHz, the size of RAM is 8.00 GB and the operating system is Windows 7. Our approach is written in Matlab.

Experimental Metrics. In this paper, we use *accuracy*, *precision*, *recall*, *specificity* and *F1-measure* to evaluate the performance of our approach. We mainly use accuracy and confusion matrix to compare the performance of our approach with other three classification approaches, which are HMM, KNN and random forest. Accuracy is the ratio of the correct number of samples to the total number of samples is correctly classified, for a given test dataset. Precision reveals the size of the random error in the classification process. Because the values of precision and recall are sometimes contradictory, we use F1-measure to solve this problem.

Experimental Results. The number of patterns we abstract in the first pattern abstraction process from all diseases is five, per signal. The total number of pattern we abstract in the first pattern abstraction process is 15. Each pattern abstracted during the first pattern abstraction process contains 120 data points. The number of atomic patterns we abstract from final pattern abstraction process is six per signal. Each atomic pattern contains 8 data points, which represent the value of the eight mentioned features.

Figure 6 shows the number of the extracted atomic patterns. In terms of the heart rate, the rate of each patterns of *sepsis* is evidently differently from

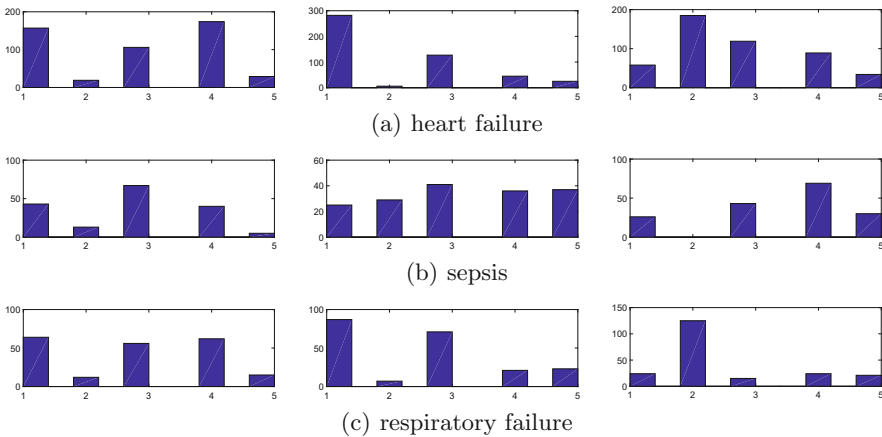


Fig. 6. The number of patterns of each disease extracted after the pattern abstraction process (*left* → *right* heart rate, blood pressure, respiratory rate).

the other two diseases. In the records of sepsis, No. 3 pattern occurs most frequently among all the heart rate patterns, while No. 1 and No. 4 pattern own the largest proportion among all the heart rate patterns in heart failure and respiratory failure. The similar situation happens in the blood pressure signals. In the respiratory rate, every pattern owns different properties in these three diseases.

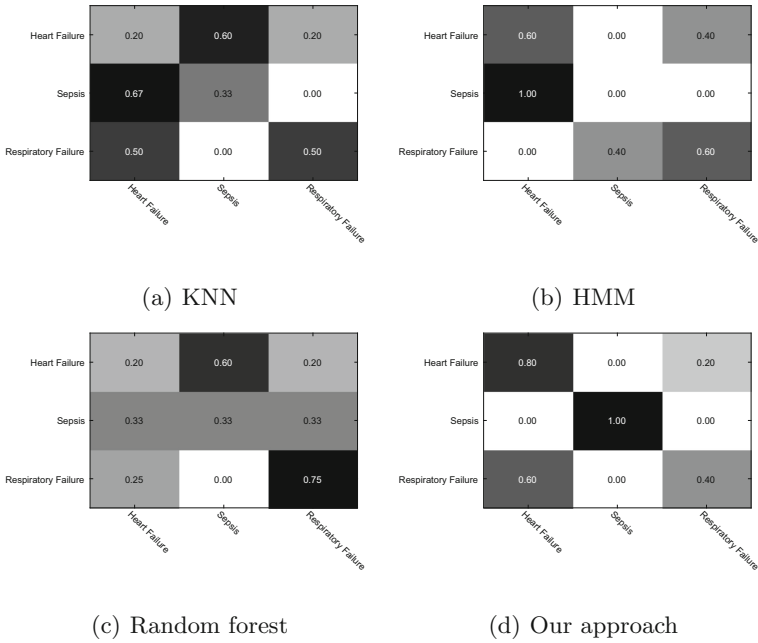


Fig. 7. Confusion matrix for different recognition approaches.

Figure 7 shows four confusion matrix of recognizing the three disease using four different classification approaches, including our approach. From the four matrix, it is obvious that our approach has the best performance for recognizing the mentioned diseases. Other three competing approaches are limited in recognizing these diseases.

Table 2. Experimental results

Disease	Precision	Recall	Sensitivity	Specificity	F1-measure
Heart failure	0.5714	0.8000	0.8000	0.6667	0.6667
Sepsis	1.0000	1.0000	1.0000	1.0000	1.0000
Resp failure	0.6667	0.5000	0.4000	0.88889	0.5000

Table 2 shows precision, recall and F1-Measure, specificity, sensitivity results over 2-fold cross-validations. The average accuracy is 0.7143. The accuracy of recognizing *Sepsis* is up to 1. The precision of recognizing respiratory failure is lower than the precision of recognizing sepsis. It is obvious that our approach is fully effective in recognizing *Sepsis*. Meanwhile our approach is less effective in recognizing *Heart Failure* and *Respiratory Failure*. This is partly because these two disease have high occurrence of comorbidity. Figure 8 shows that the accuracy of recognising the three diseases using our approach outperforms using KNN, HMM and Random Forest.

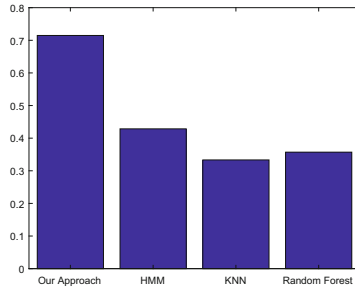


Fig. 8. Accuracy comparison of the four competing approaches.

7 Conclusion

In this paper, we present an approach combining pattern abstraction and Probabilistic Interval-Based Bayesian Model where Allen's relations are used to depict the relations between two intervals within signals of diseases to recognize diseases. From our experimental results, especially the accuracy of recognizing certain disease is up to 1, we can conclude that our approach is capable to recognize diseases from physiological signals. Also, it is more efficient and flexible than existing methods including KNN for disease recognition. In the future, we will increase the types of diseases to increase the general applicability of the approach.

Acknowledgements. This work was supported by grants from the Fundamental Research Funds for the Key Research Programm of Chongqing Science & Technology Commission (grant no. cstc2017rgzn-zdyf0064), the Chongqing Provincial Human Resource and Social Security Department (grant no. cx2017092), the Central Universities in China (grant nos. CQU0225001104447).

References

1. Allen, J.F.: Maintaining knowledge about temporal intervals. *ACM* (1983)
2. Banaee, H., Loutfi, A.: Data-driven rule mining and representation of temporal patterns in physiological sensor data. *IEEE J. Biomed. Health Inform.* **19**(5), 1557–1566 (2015)
3. Beumer, M.: Qualitative probabilistic networks in medical diagnosis (2006)
4. Fatima, M., Pasha, M.: Survey of machine learning algorithms for disease diagnostic. *J. Intell. Learn. Syst. Appl.* **01**(1), 1–16 (2017)
5. Fu, T.C.: A review on time series data mining. *Eng. Appl. Artif. Intell.* **24**(1), 164–181 (2011)
6. Goldin, D., Mardales, R., Nagy, G.: In search of meaning for time series subsequence clustering: matching algorithms based on a new distance measure, pp. 347–356 (2006)
7. He, J., et al.: An association rule analysis framework for complex physiological and genetic data. *J. Solid State Chem.* **220**, 185–190 (2012)
8. Johnson, A.W.E., et al.: MIMIC-III, a freely accessible critical care database. *Scientific Data* **3**, 160035 (2016)
9. Liu, L., Cheng, L., Liu, Y., Jia, Y., Rosenblum, D.S.: Recognizing complex activities by a probabilistic interval-based model. In: *National Conference on Artificial Intelligence* (2016)
10. Marlin, B.M., Kale, D.C., Khemani, R.G., Wetzell, R.C.: Unsupervised pattern discovery in electronic health care data using probabilistic clustering models. In: *Proceedings of the 2nd ACM SIGHIT International Health Informatics Symposium*, pp. 389–398 (2012)
11. Muffikhah, L., Wahyuningsih, Y., Nbsp, M.: Fuzzy rule generation for diagnosis of coronary heart disease risk using subtractive clustering method. *J. Softw. Eng. Appl.* **06**(07), 372–378 (2013)
12. Ni, J., Fei, H., Fan, W., Zhang, X.: Cross-network clustering and cluster ranking for medical diagnosis. In: *IEEE International Conference on Data Engineering*, pp. 163–166 (2017)
13. Nikovski, D.: Constructing Bayesian networks for medical diagnosis from incomplete and partially correct statistics. *IEEE Trans. Knowl. Data Eng.* **12**(4), 509–516 (2000)
14. Nisha, S., Kathija, A.: Breast cancer data classification using SVM and Naive Bayes techniques. *International J. Innov. Res. Comput. Commun. Eng.* **4**(12) (2016)
15. Pitman, J.: Combinatorial stochastic processes. Technical report 621, Department of Statistics, UC Berkeley, Lecture notes (2002)
16. Sacchi, L., Bellazzi, R., Larizza, C., Porreca, R., Magni, P.: Learning rules with complex temporal patterns in biomedical domains. In: Miksch, S., Hunter, J., Keravnou, E.T. (eds.) *AIME 2005. LNCS (LNAI)*, vol. 3581, pp. 23–32. Springer, Heidelberg (2005). https://doi.org/10.1007/11527770_4
17. Zhang, Y., Zhang, Y., Swears, E., Larios, N., Wang, Z., Ji, Q.: Modeling temporal interactions with interval temporal Bayesian networks for complex activity recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **35**(10), 2468–2483 (2013)

Knowledge Engineering Applications



An Incremental Approach Based on the Coalition Formation Game Theory for Identifying Communities in Dynamic Social Networks

Qing Xiao, Peizhong Yang, Lihua Zhou (✉), and Lizhen Wang

School of Information, Yunnan University, Kunming 650500, China
{Xiaoqing, lhzhou, lzhwang}@ynu.edu.cn,
285342456@qq.com

Abstract. Most real-world social networks are usually dynamic (evolve over time), thus communities are constantly changing in memberships. In this paper, an incremental approach based on the coalition formation game theory to identify communities in dynamic social networks is proposed, where the community evolution is modeled as the problem of transformations of stable coalition structures. The proposed approach adaptively update communities from the previous known structures and the changes of topological structure of a network, rather than re-computing in the snapshots of the network at different time steps, such that the computational cost and processing time can be significantly reduced. Experiments have been conducted to evaluate the effectiveness of the proposed approach.

Keywords: Dynamic social network · Incremental community detection
Coalition formation game theory · D_c -stable partitioning

1 Introduction

In social networks, community detection means to divide nodes of a network into groups such that nodes in each group are densely connected inside and sparser outside (Newman and Girvan 2004; Fortunato 2010). It is an important research topic in social networks analysis and has been used in many real applications, such as structure visualizing (Wu and Li 2011), time series clustering (Ferreira and Zhao 2016).

Unfortunately, community detection in social networks is challenging, especially in dynamic social networks where topological structures of networks evolve with time. The dynamics of the network over a long period of time may significantly transform the current community structure to a totally different one (Nguyen et al. 2011). The community evolution raises a natural need of re-identification of communities in updated networks. Although the multiple execution of any community detection algorithm designed for static networks without topological changes, such as LM method (Blondel et al. 2008), can be applied to find the new community structure whenever the network evolves, the long running time of a specific static method on large networks cannot be neglected. Furthermore, the trap of local optima and the

almost same reaction to a small change to some local part of the network are also taken into consideration (Nguyen et al. 2011), thus the research on fast approach to identify communities in dynamic social networks is very important and necessary.

In this paper, we propose an incremental approach to identify communities in dynamic social networks based on coalition formation game theory (Saad et al. 2009), whose main objective is to analyze the formation of coalitional structures through players' interaction. In our approach, a dynamic network is represented by a sequence of network snapshots evolving over time, and the network communities can be adaptively updated from the previous known structures and the change of topological structures of networks, rather than re-computing in the snapshots network at different time steps.

In summary, the specific contributions of this paper are highlighted as follows:

- (1) The stability of a coalition structure is discussed and the community evolution is modeled as the problem of transformations of stable coalition structures;
- (2) The effects of the change of topological structures of a network on the communities of the network is analyzed, and the method for detecting communities based on the known stable coalition structures and the change of topological structures of a network is proposed;
- (3) Experiments have been conducted to evaluate the effectiveness of the proposed approach.

The rest of this paper is organized as follows: Sect. 2 introduces related work; Sect. 3 discusses the stability of a community structure; Sect. 4 analyzes how a community structure is affected by changes of a network. The experimental results are presented in Sect. 5, and Sect. 6 concludes this paper.

2 Related Work

Community detection in dynamic networks has attracted much attention in recent years and many methods have been proposed. Nguyen et al. (2011) proposed an adaptive modularity-based approach (QCA). This approach has not only the power of quickly and efficiently updating the network communities through a series of changes, by only using the structures identified from previous network snapshots, but also the ability of tracing the evolution of community structure over time. Dinh et al. (2015) presented an adaptive framework with approximation guarantees (A3CS). Hecking et al. (2014) extended an existing method for optimizing modularity in unipartite networks to dynamic bipartite networks for identifying clusters in evolving bipartite networks over time. Shang et al. (2016) used machine learning classifiers to predict the vertices that need to be inspected for community assignment revision. Lee et al. (2016) proposed an algorithm for updating betweenness centrality in fully dynamic graphs and adapted a community detection algorithm using the proposed algorithm. Different from the works that focus on community detection in dynamic networks, Du et al. (2015) proposed a framework to track the progression of the community strength that reflects the community robustness and coherence throughout the entire observation period. It is effective in discovering the progression of community strengths and detecting interesting communities.

Game theories have been used to solve community detection problems. For example, Chen et al. (2010); Alvari et al. (2014) addressed the overlapping community detection problem in static networks by a non-cooperative game theory-based framework. Zhou et al. (2015) proposed a coalition formation game theory-based approach to detect overlapping and hierarchical communities, but this approach is just appropriate for static networks.

3 The Stability of a Community Structure

3.1 The Formation of a Coalition Structure and Its Stable Conditions

Let N be a fixed set of players, non-empty subsets $S_i \subset N$ are called coalitions, a collection of coalitions in N is any family $\Gamma = \{S_1, S_2, \dots, S_k\}$ of mutually disjoint coalitions. If additionally $\bigcup_{j=1}^k S_j = N$, the collection $\Gamma = \{S_1, S_2, \dots, S_k\}$ is called a partitioning or a coalition structure of N . Let $v(S_i)$ be the utility of S_i , $\Gamma' = \{S'_1, S'_2, \dots, S'_l\}$ be a coalition structure different from Γ , if $\sum_{i=1}^k v(S_i) > \sum_{i=1}^l v(S'_i)$, then a group of players prefers to organize themselves into a collection $\Gamma = \{S_1, S_2, \dots, S_k\}$ instead of $\Gamma' = \{S'_1, S'_2, \dots, S'_l\}$ (Apt and Witzel 2009). Given a set of players N , any collection of disjoint coalitions $\{S_1, S_2, \dots, S_k\}, S_i \subset N$ can agree to merge into a single coalition $S = \bigcup_{i=1}^k S_i$, if $v(S) > \sum_{i=1}^k v(S_i)$; a coalition S splits into smaller coalitions if $v(S) < \sum_{i=1}^k v(S_i)$. A stable coalition structure is called a stable community structure.

In various coalition structures, the D_c -stable partitioning maximizes the total utility, and no groups of players in a D_c -stable partitioning have incentive to leave this partitioning for forming any other collection in N , i.e. the players prefer the D_c -stable partitioning over all other partitions. Theorem 1 (Apt and Witzel 2009) gave the conditions needed for the existence of D_c -stable partitioning.

Theorem 1 (Apt and Witzel 2009). A partitioning $\Gamma = \{S_1, S_2, \dots, S_k\}$ of N is D_c -stable iff the following two conditions are satisfied:

- (i) for each $i \in \{1, \dots, k\}$ and each pair of disjoint coalition A and B such that $A \cup B \subseteq S_i, \{A \cup B\} \triangleright \{A, B\}$;
- (ii) for each Γ - incompatible coalition $T \subseteq N, \{T\}[T] \triangleright \{T\}$.

Where T is a coalition, Γ - incompatible means for some $i \in \{1, \dots, k\}$ exist $T \not\subseteq S_i, \{T\}[T] = \{S_1 \cap T, \dots, S_k \cap T\} \setminus \{\varnothing\}$ denote a coalition collection; \triangleright be a comparison relation, $\Gamma \triangleright \Gamma'$ means that the way Γ partitions N is preferable to the way Γ' partitions N .

3.2 The Stability of a Community Structure

In social network environments, the behaviors' of individuals are not independent (Zacharias et al. 2008), and joining a community provides one with tremendous benefits, such as members feeling rewarded in some ways for their participation in the community, and gaining honors and status for being members (Sarason 1974). In which

case, every individual has an incentive to join communities; however, in real-world cases not only does each individual receive benefit(s) from the communities it belongs to, but the individual must also pay a certain price to maintain its membership within these communities (Chen et al. 2010). These characteristics make community detection problem can be solved by coalition formation game theory.

Let $G = (N, E)$ be an undirected unweighted graph representing a social network with n nodes and m links. Let A be an adjacency matrix of G , and $d(x)$ be the degree of node x . Let S denote a subset of N , which is called a coalition, meanwhile let $e(S)$, $d(S)$ and $v(S)$ be the number of links amongst nodes inside S , the total degree of nodes in S and the utility function of S , respectively. Let $e(x, S)$ be the number of edges in $G = (N, E)$ that link the node x to nodes of coalition S , i.e. $e(x, S) = \sum_{y \in S, y \neq x} A(x, y)$. For

any coalition $S_1, S_2 \subseteq N$, let $e(S_1, S_2)$ be the number of links connecting nodes of the coalition S_1 to the nodes of the coalition S_2 .

A utility function $v(S)$ for coalition S and the total utility $v(\Gamma)$ of a coalition structure $\Gamma = \{S_1, S_2, \dots, S_k\}$ is defined as $v(S) = \frac{2e(S)}{d(S)} - \alpha \left(\frac{d(S)}{2\beta m}\right)^2$ and $v(\Gamma) = \sum_{S_i \in \Gamma} v(S_i)$ (Zhou et al. 2015). Based on the definition of $v(S)$, Theorem 2 gives the conditions needed for the existence of a D_c -stable coalition structure.

Theorem 2. A coalition structure $\Gamma = \{S_1, S_2, \dots, S_k\}$ is approximate D_c -stable if $\forall S_i \in \Gamma, \forall T \subset S_i, v(S_i) > v(T)$, and for all $x \notin S_j, \frac{e(x, S_j)}{d(x)} < \frac{e(S_j)}{d(S_j)}$.

The proof is omitted by the limitation of space. In general, forming a coalition structure is a dynamic process and a D_c -stable coalition structure can be obtained by merge and split iteration. Before entering a stable state, some nodes have incentives to change their coalition memberships, i.e. leave from their current coalitions or join in other coalitions with the changes of game environment. Lemma 1 gives the condition needed for merging two coalitions into a larger coalition, and Lemma 2 gives the condition that a node leaves from or joins in a coalition.

Lemma 1. Given a coalition structure $\Gamma = \{S_1, S_2, \dots, S_k\}$, for any $S_i, S_j \in \Gamma, S_i$ merges with S_j into $S_i + S_j$ if

$$e(S_i, S_j) > \frac{e(S_i)}{d(S_i)}d(S_j) - e(S_j) + \alpha \frac{2d(S_i)d(S_j) + d^2(S_j)}{8\beta^2 m^2} [d(S_i) + d(S_j)] \ \&\&$$

$$e(S_i, S_j) > \frac{e(S_j)}{d(S_j)}d(S_i) - e(S_i) + \alpha \frac{2d(S_i)d(S_j) + d^2(S_i)}{8\beta^2 m^2} [d(S_i) + d(S_j)].$$

Corollary 1. Given a coalition structure $\Gamma = \{S_1, S_2, \dots, S_k\}, S_i, S_j \in \Gamma, S_i \neq S_j, S_i$ can not merge with S_j to form a larger coalition if $e(S_i, S_j) = 0$.

Based on the Corollary 1, whether a coalition is merged with others can be decided by looking only at its neighbors (coalitions that have links between them), without an exhaustive search over the entire network.

Lemma 2. Given a coalition structure $\Gamma = \{S_1, S_2, \dots, S_k\}$, for any $S_i, S_j \in \Gamma$, if $x \in S_i$ & $\frac{e(x, S_i)}{d(x)} < \frac{e(S_i)}{d(S_i)}$, x leaves from S_i ; if $x \notin S_j$ & $\frac{e(x, S_j)}{d(x)} > \frac{e(S_j)}{d(S_j)} + \frac{\alpha}{2} \left(\frac{2d(S_j) + d(x)}{2\beta m} \right)^2$, x joins in S_j .

Based on the Lemma 2, we can give the Corollaries 2, 3 and the Lemma 3.

Corollary 2. Given a coalition structure $\Gamma = \{S_1, S_2, \dots, S_k\}$, $S_j \in \Gamma$, $x \notin S_j$. x can not join in S_j if $e(x, S_j) = 0$.

Corollary 3. Given a coalition structure $\Gamma = \{S_1, S_2, \dots, S_k\}$, $S_i, S_j \in \Gamma$, $S_i \neq S_j$. The coalition membership of x will be intact if $x \in S_i$ & $\frac{e(x, S_i)}{d(x)} > \frac{e(S_i)}{d(S_i)} + \frac{\alpha}{2} \left(\frac{2d(S_i) + d(x)}{2\beta m} \right)^2$, or $x \notin S_j$ & $\frac{e(x, S_j)}{d(x)} < \frac{e(S_j)}{d(S_j)}$. If $\Gamma = \{S_1, S_2, \dots, S_k\}$ is a D_c -stable coalition structure, $\frac{e(x, S_i)}{d(x)} > \frac{e(S_i)}{d(S_i)}$ for all $x \in S_i$ and $\frac{e(x, S_j)}{d(x)} < \frac{e(S_j)}{d(S_j)}$ for all $x \notin S_j$.

Lemma 3. Given a coalition structure $\Gamma = \{S_1, S_2, \dots, S_k\}$, $S_i, S_j \in \Gamma$, $S_i \neq S_j$, $x \in S_i$. If x is an internal node of S_i , i.e. $e(x, S_i) = d(x)$, then x neither leaves from S_i nor joins in S_j .

4 The Approach for Detect Communities in Dynamic Social Networks

In this section, we first give the notations of dynamic social networks and the problem definition, and then we analyze how a community structure is affected by changes of a network.

4.1 The Notations of Dynamic Social Networks and the Problem Definition

Let $G^t = (N^t, E^t)$ be a time dependent network snapshot recorded at time t , $\Delta G^t = (\Delta N^t, \Delta E^t)$ be the change of topological structures of networks, where ΔN^t and ΔE^t be the sets of nodes and links to be introduced (or removed) in the period $[t, t + 1)$. The next network snapshot G^{t+1} is the current one together with changes, i.e., $G^{t+1} = G^t \cup \Delta G^t$. A dynamic network \mathbf{G} is a sequence of network snapshots evolving over time: $\mathbf{G} = (G^1, G^2, \dots, G^t)$.

Problem Definition: Given a dynamic network $\mathbf{G} = (G^1, G^2, \dots, G^t)$ where G^1 is the initial network and G^2, G^3, \dots, G^t are the network snapshots obtained through $\Delta G^1, \Delta G^2, \dots, \Delta G^{t-1}$, we need to devise an incremental algorithm to efficiently detect and identify the D_c -stable coalition structure at any time point, utilizing the information from the previous snapshots as well as tracing the evolution of the network, i.e. finding G^{t+1} based on G^t and ΔG^t .

Nguyen et al. (2011) observed that the introduction or removal a set of nodes (or edges) can be decomposed as a sequence of node (or edge) insertions (or removals), in which a single node (or a single edge) is introduced (or removed) at a time. Based on

these observations, they defined following four simple events to reflect changes introduced to a social network:

- *NewNode* ($N + \{x\}$): A new node x with its associated edges are introduced. x could come with no or more than one new edge(s).
- *NewEdge* ($E + \{(x, y)\}$): A new edge (x, y) connecting two existing nodes x and y is introduced.
- *RemoveEdge* ($E - \{(x, y)\}$): An existing edge (x, y) in a network is removed.
- *RemoveNode* ($N - \{x\}$): A node x and its adjacent edges are removed from a network.

In this paper, we also use these events to reflect changes introduced to a social network and treat network changes as a collection of simple events *NewNode*, *NewEdge*, *RemoveEdge*, *RemoveNode*. We model the community evolution as the problem of transformations of stable coalition structures. Specifically, the first stable coalition structure Γ^1 can be detected in the first snapshots network G^1 by using a static community detection algorithm, but the stability of Γ^1 may be broken by any one of simple events, for example, two coalitions in Γ^1 having less distraction caused by each other may be combined to form a new coalition due to the introduction of *newEdge*, thus Γ^1 need to update to enter a new stable state. Γ^2 can be obtained by a series of updates of Γ^1 .

Next, we analyze how a stable coalition structure is affected by simple events and give our incremental methods for updating coalitions.

4.2 Effects of Simple Events on a Stable Coalition Structure and the Incremental Methods for Updating Coalitions

4.2.1 *NewNode*—A New Node x and Its Associated Connections Are Introduced

- (1) x has no adjacent edge, then a new coalition containing only x is created and the other coalitions remain intact.
- (2) x comes with edges connecting one or more existing coalitions. In this case, we need to determine which coalition x should join in. By Lemma 2, joining the new node x in coalition S_j will increase the utility value $v(S_j)$ if $\frac{e(x, S_j)}{d(x)} > \frac{e(S_j)}{d(S_j)} + \frac{\alpha}{2} \left(\frac{2d(S_j) + d(x)}{2\beta m} \right)^2$. Thus, if this inequality holds, then it is reasonable for x joining in coalition S_j . If x can not join any one coalition, then a new coalition containing only x is created.

Example 1. In Fig. 1(a), p joins S_2 , in Fig. 1(b), p forms a new coalition, and S_1 merges with S_2 into S'_1 in Fig. 1(c).

4.2.2 *NewEdge*—A New Edge $l = (x, y)$ Connecting Two Existing Nodes x and y Is Introduced

In this paper, the term *intra-coalition links* refers to edges whose two endpoints belong to the same coalition, while the term *inter-coalition links* refers to those with endpoints

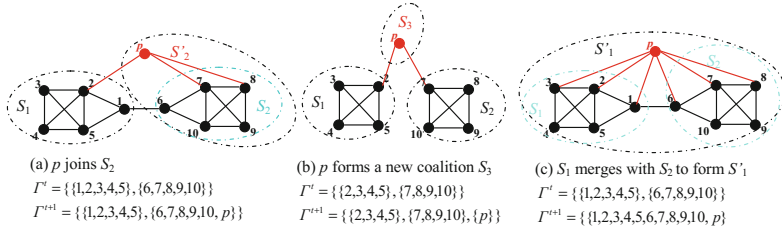


Fig. 1. A new node p is introduced with associated links

connecting different coalitions. In order to deal with the introduction of an intra-coalition link, we first give Lemma 4 as follows.

Lemma 4. Given a D_C -stable coalition structure $\Gamma^t = \{S_1, S_2, \dots, S_k\}$, $S_i, S_j \in \Gamma^t, S_i \neq S_j, x, y \in S_i$. Then except for the boundary nodes of S_i that satisfy $\frac{e(z, S_i)}{d(z)} \leq \frac{e(S_i) + 1}{d(S_i) + 2}, z \in S_i$ may leave their current coalitions, other nodes in S_i and nodes in S_j have no incentive to leave from its current coalition or to join in other coalitions after $l = (x, y)$ is introduced.

- (1) $l = (x, y)$ is an intra-coalition link ($x, y \in S_i$). According to Lemma 4, S_i is updated as $S_i - \{z\}$ if $z \in S_i$ & $\frac{e(z, S_i)}{d(z)} \leq \frac{e(S_i) + 1}{d(S_i) + 2}$, other coalitions remain intact; otherwise the current coalition structure remains intact. Intuitively, the introduction of an intra-coalition link will strengthen the inner structure of coalition S_i , thus the current coalition structure remains intact.
- (2) $l = (x, y)$ is an inter-coalition link ($x \in S_i, y \in S_j, S_i \neq S_j$). Its presence may make S_i merges with S_j into a larger coalition.

Example 2. In Fig. 2(a), $l = (p, 6)$ is an intra-coalition link while the added edge is an inter-coalition link in Fig. 2(b). In Fig. 2(a), the current coalition structure does not affected by the introduce of $l = (p, 6)$, but in Fig. 2(b), the introduce of $l = (p, 3)$ makes the coalition S_1 merges with S_2 .

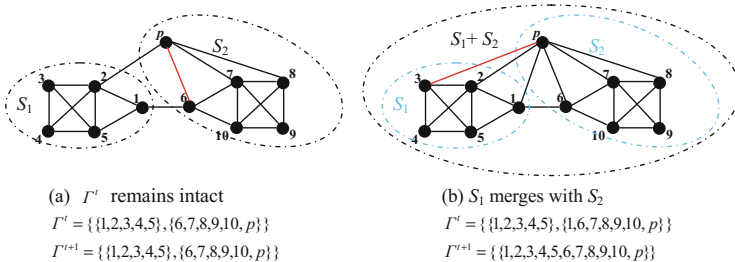


Fig. 2. A new edge (x, y) is introduced

4.2.3 RemoveEdge—An Existing Edge $l = (x, y)$ Is Removed

The removed edge $l = (x, y)$ may be an intra-coalition link or an inter-coalition links. To deal with the introduction of an intra-coalition link, we first give Lemma 5 as follows.

Lemma 5. Given a D_c -stable coalition structure $\Gamma^t = \{S_1, S_2, \dots, S_k\}$, $S_i, S_j \in \Gamma^t$, $S_i \neq S_j$, $x \in S_i$, $y \in S_j$. Then except for the boundary nodes of S_i and S_j that satisfy $\frac{e(z, S_i)}{d(z)} \leq \frac{e(S_i)}{d(S_i)-1}$, $z \in S_i$ and $\frac{e(p, S_j)}{d(p)} \leq \frac{e(S_j)}{d(S_j)-1}$, $p \in S_j$ may leave their current coalitions, other nodes in S_i and nodes in S_j have no incentive to leave from its current coalition or to join in other coalitions after $l = (x, y)$ is removed.

- (1) $l = (x, y)$ is an inter-coalition link ($x \in S_i, y \in S_j, S_i \neq S_j$). According to Lemma 5, S_i is updated as $S_i - \{z\}$ if $z \in S_i$ & $\frac{e(z, S_i)}{d(z)} \leq \frac{e(S_i)}{d(S_i)-1}$, S_j is updated as $S_j - \{p\}$ if $p \in S_j$ & $\frac{e(p, S_j)}{d(p)} \leq \frac{e(S_j)}{d(S_j)-1}$, other coalitions remain intact; otherwise the current coalition structure remains intact. Intuitively, the removal of an inter-coalition link will strengthen the coalition S_i and S_j , hence the current coalition structure remains intact.
- (2) $l = (x, y)$ is an intra-coalition link ($x, y \in S_i$). This case can be divided further into three subcases:
 - x and y are nodes of degree one. Then, S_i splits into two coalitions containing isolate node x and y respectively, while the other coalitions remain intact.
 - Only one of x or y has degree one (Let $d(x) > 1$, and $d(y) = 1$ for simplicity). Then, S_i splits into two coalitions, one consists of y , the other consists of nodes of $S_i - \{y\}$. The other coalitions remain intact.
 - Both x and y are nodes with degrees greater than one. In this subcase, S_i may be either unchanged or broken into smaller sub-coalitions and some of them could probably be merged with other coalitions.

To deal with the case that both x and y are nodes with degrees greater than one, we first identify the leftover structure of S_i after $l = (x, y)$ is removed by using a static community detection algorithm, then detect whether coalitions in the leftover structure can merge with coalitions except for S_i ; at last we detect whether boundary nodes of sub-coalitions and their neighbors change their coalition memberships.

Example 3. In Fig. 3(a), $l = (1, y)$ where y has degree one, and the remove of $l = (1, y)$ makes node 1 leaves from S_1 ; In Fig. 3(b), the deletion of $l = (3, p)$ makes the coalition S_1 split into two coalitions.

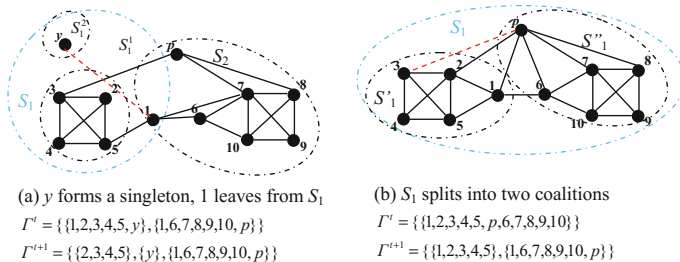


Fig. 3. An edge is deleted

4.2.4 RemoveNode—An Existing Node $x \in S_i$ and Its Associated Connections Are Removed

When an existing node $x \in S_i$ is removed at time t , all of its adjacent edges are removed.

- (1) x is a singleton node. Then only the coalition containing x will be removed and the other coalitions leave intact.
- (2) x is a node with degree one. Then the host coalitions containing x are updated by removing x and the other coalitions also leave intact.
- (3) x is a node with degree greater than one. In this case, S_i may be either unchanged or broken into smaller sub-coalitions and some of them could probably be merged with other coalitions.

In the third case, we first identify the leftover structure of S_i , then detect whether coalitions in the leftover structure can merge with coalitions except for S_i ; at last we detect whether boundary nodes of sub-coalitions and their neighbors change their coalition memberships.

Example 4. In Fig. 4(a), the deletion of node 4 incurs the merger of S_1 and S_2 . In Fig. 4(b), the deletion of node 2 incurs node p leaves S_1 and node 1 joins in S_2 .

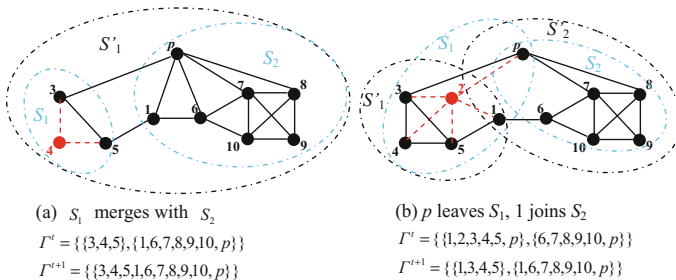


Fig. 4. A node is deleted

5 Experiments and Results

The real dynamic network used in this paper is the Enron email network constructed from the Enron email dataset (Sun et al. 2007). This dataset includes email messages data from about 150 users, mostly senior management of Enron Inc. from January 1999 to July 2002. After cleaning and refining the data, we construct a dynamic network with 20 network snapshots, where each node represents a user and each edge represents an interaction of sending or receiving email between two users. In the first snapshot, 30% of total edges amongst 150 nodes are selected randomly to form a basic community structure of the network with 7 major communities, and then about 3.7% of total edges are added from G^t to G^{t+1} , $t = 0, 1, \dots, 19$. The evaluation metrics include *modularity* (Newman and Girvan 2004), running time, number of communities and the *normalized mutual information* (NMI) (Danon et al. 2005).

We compare the performance of our method (denoted as *COFOGAInc*) with four state-of-the-art baseline algorithms: *Louvain algorithm* (Blondel et al. 2008), *BatchInc algorithm* (Chong and Teow 2013), *QCA* (Nguyen et al. 2011) and *LBTR algorithm* (Shang et al. 2016). The modularity values, numbers of communities, running time and NMI scores computed by *Louvain*, *BatchInc*, *QCA*, *LBTR* and *COFOGAInc* method are shown in Fig. 5(a), (b), (c) and (d) respectively. In Fig. 5(d), the community structure identified by *Louvain algorithm* is used as a reference to the ground truth for computing NMI scores of other algorithms, because the proper information about real communities in this dataset is lacked.

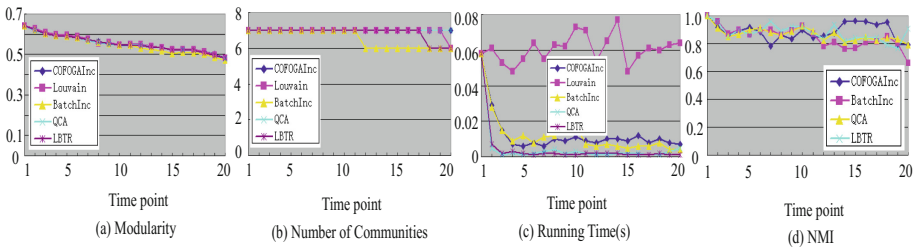


Fig. 5. The results of algorithms on the *Enron* email network

From Fig. 5, we can see that the modularity values computed by five algorithms are relatively close; the numbers of communities maintain the same by *COFOGAInc* and *QCA algorithm* but with a little variation by *Louvain*, *BatchInc* and *LBTR algorithm* in all snapshots; the running times of *COFOGAInc* and *BatchInc* are lesser than the one that *Louvain algorithm* consumes but are more than those that *QCA* and *LBTR algorithm* take; the NMI scores between *COFOGAInc*, *BatchInc*, *QCA*, *LBTR* and *Louvain algorithm* are very high and relatively close to 1, indicating that the community labels assigned by *COFOGAInc*, *BatchInc*, *QCA*, *LBTR algorithm* are similar to those assigned by *Louvain algorithm*.

The results of Fig. 5 indicate that *COFOGAInc* algorithm proposed in this paper is able to identify high quality community structure (with high *modularity* value and high NMI score), and the incremental way can reduce the detecting time.

6 Conclusion

In this paper, we propose an incremental approach based on the coalition formation game theory to identify communities in dynamic social networks. In our approach, a dynamic network is represented as a sequence of network snapshots evolving over time, and the community evolution is model as the problem of transformations of stable coalition structures. We present the conditions needed for the existence of a D_c -stable coalition structure and present approaches for updating coalitions. The proposed approach adaptively update network communities from the previous known structures and the change of topological structures of networks, rather than re-computing in the

snapshots network at different time steps, such that the computational cost and processing time can be significantly reduced.

Acknowledgement. This research was supported by the National Natural Science Foundation of China (61762090, 61262069, 61472346, and 61662086), The Natural Science Foundation of Yunnan Province (2016FA026, 2015FB114), the Project of Innovative Research Team of Yunnan Province, and Program for Innovation Research Team (in Science and Technology) in University of Yunnan Province (IRTSTYN).

References

- Alvari, H., Hajibagheri, A., Sukthakar, G.: Community detection in dynamic social networks: a game-theoretic approach. In: ASONAM, China, 17–20 August 2014, pp. 101–107 (2014)
- Apt, K.R., Witzel, A.: A generic approach to coalition formation. *Int. Game Theory Rev.* **11**(03), 347–367 (2009)
- Chen, W., Liu, Z., Sun, X., Wang, Y.: A game-theoretic framework to identify overlapping communities in social networks. *Data Mining Knowl. Discov.* **21**(2), 224–240 (2010)
- Chong, W.H., Teow, L.N.: An incremental batch technique for community detection. In: FUSION, Istanbul, Turkey, 9 July–12 July 2013, pp. 750–757 (2013)
- Blondel, V.D., Guillaume, J.L., Lambiotte, R., Lefebvre, E.: Fast unfolding of communities in large networks. *J. Stat. Mech: Theory Exp.* **2008**, P10008 (2008)
- Danon, L., Díaz-Guilera, D.A., Duch, J., Arenas, A.: Comparing community structure identification. *J. Stat. Mech: Theory Exp.* **2005**(09), P09008 (2005)
- Du, N., Jia, X., Gao, J., Gopalakrishnan, V., Zhang, A.: Tracking temporal community strength in dynamic networks. *IEEE Trans. Knowl. Data Eng.* **27**(11), 3125–3137 (2015)
- Dinh, T.N., Nguyen, N.P., Alim, M.A., Thai, M.T.: A near-optimal adaptive algorithm for maximizing modularity in dynamic scale-free networks. *J. Comb. Optim.* **30**(3), 747–767 (2015)
- Ferreira, L.N., Zhao, L.: Time series clustering via community detection in networks. *Inf. Sci.* **326**(1), 227–242 (2016)
- Fortunato, S.: Community detection in graphs. *Phys. Rep.* **486**, 75–174 (2010)
- Hecking, T., Steinert, L., Gohnert, T., Hoppe, H.U.: Incremental clustering of dynamic bipartite networks. In: ENIC, Wroclaw, Poland, 29–30 September 2014, pp. 9–16 (2014)
- Lee, M., Choi, S., Chung, C.W.: Efficient algorithms for updating betweenness centrality in fully dynamic graphs. *Inf. Sci.* **326**(2016), 278–296 (2016)
- Newman, M.E.J., Girvan, M.: Finding and evaluating community structure in networks. *Phys. Rev. E* **69**, 026113 (2004)
- Nguyen, N.P., Dinh, T.N., Xuan, Y., Thai, M.T.: Adaptive algorithms for detecting community structure in dynamic social networks. In: INFOCOM, Shanghai, China, 10–15 April 2011, pp. 2282–2290 (2011)
- Saad, W., Han, Z., Debbah, M., Hjørungnes, A., Basar, T.: Coalitional game theory for communication networks: a tutorial. *IEEE Sig. Process. Mag.* **26**(5), 77–97 (2009)
- Sarason, S.B.: *The Psychological Sense of Community: Prospects for a Community Psychology*. Jossey-Bass, San Francisco (1974)
- Shang, J., Liu, L., Li, X., Xie, F., Wu, C.: Targeted revision: a learning-based approach for incremental community detection in dynamic networks. *Phys. A: Stat. Mech. Appl.* **443** (2016), 70–85 (2016)

- Sun, J., Faloutsos, C., Papadimitriou, S., Yu, P.S.: Graphscope: parameter-free mining of large time-evolving graphs. In: SIGKDD, San Jose, USA, 11–14 August 2007, pp. 687–696 (2007)
- Wu, P., Li, S.K.: Social network analysis layout algorithm under ontology model. *J. Softw.* **6**(7), 1321–1328 (2011)
- Zacharias, G.L., MacMillan, J., Hemel, S.B.V. (eds.): Behavioral Modeling and Simulation: From Individuals to Societies. National Academies Press, Washington, DC (2008)
- Zhou, L., Lü, K., Yang, P., Wang, L., Kong, B.: An approach for overlapping and hierarchical community detection in social networks based on coalition formation game theory. *Expert Syst. Appl.* **42**(24), 9634–9646 (2015)



LogRank: An Approach to Sample Business Process Event Log for Efficient Discovery

Cong Liu¹, Yulong Pei², Qingtian Zeng¹(✉), and Hua Duan¹(✉)

¹ Shandong University of Science and Technology, Qingdao, China
{liucongchina,qtzeng,hduan}@sdust.edu.cn

² Eindhoven University of Technology, Eindhoven, The Netherlands
y.pei.1@tue.nl

Abstract. Considerable amounts of business process event logs can be collected by modern information systems. Process discovery aims to uncover a process model from an event log. Many process discovery approaches have been proposed, however, most of them have difficulties in handling large-scale event logs. Motivated by *PageRank*, in this paper we propose *LogRank*, a graph-based ranking model, for event log sampling. Using *LogRank*, a large-scale event log can be sampled to a smaller size that can be efficiently handled by existing discovery approaches. Moreover, we introduce an approach to measure the quality of a sample log with respect to the original one from a discovery perspective. The proposed sampling approach has been implemented in the open-source process mining toolkit ProM. The experimental analyses with both synthetic and real-life event logs demonstrate that the proposed sampling approach provides an effective solution to improve process discovery efficiency as well as ensuring high quality of the discovered model.

Keywords: LogRank · Log sampling · Process discovery
Quality measure

1 Introduction

Process mining [1, 10, 20] is an active research discipline aiming at extracting insights about business processes from event logs. Process discovery allows to distil process models from event logs. Researchers have proposed various process discovery approaches that take an event log as input and produce a process model without using any priori information. However, most of existing process discovery approaches cannot handle properly or may cause low efficiency when facing large-scale event logs.

Given a large-scale event log, one effective strategy is to re-implement some discovery approaches using MapReduce to make them scalable to large data sets. *Evermann* [7] presents the MapReduce implementations of the *Alpha Miner* and *Heuristic Miner*. Rather than re-implementing existing discovery approaches, we

propose to sample the large-scale event log to a manageable size that can be efficiently handled by existing discovery approaches in the paper. Theoretically, we can select an arbitrary subset of traces from an event log as its sample log. The real challenge is how to find a sample log that is representative enough to discover a reliable process model compared with the original event log.

To this end, we propose *LogRank* - a graph-based ranking model to obtain a representative sample log by taking an arbitrary (big) event log as input. In addition, we also introduce an approach to measure the quality of a sample log with respect to the original one from the discovery perspective. To the best of our knowledge, this is the first paper that tries to improve process discovery efficiency by sampling large-scale event logs.

The rest of this paper is organized as follows. Section 2 defines some preliminaries. Section 3 presents the research questions and introduces an overview of our approach. Section 4 introduces the *LogRank*-based approach to sample an event log and the quality measurement of the sample log. Section 5 presents tool support and experimental evaluation. Finally, Sect. 6 concludes the paper.

2 Preliminaries

Let S be a set. We use $|S|$ to denote the number of elements in set S . $\mathbf{B}(S)$ is the set of all multisets over set S . $f \in X \rightarrow Y$ is a function, i.e., $dom(f)$ is the domain and $rng(f) = \{f(x) | x \in dom(f)\}$ is the range.

Definition 1 (Event, Trace, Event Log). Let A be a set of activities. A trace $\sigma \in A^*$ is a sequence of activities (also referred to as events). For $1 \leq i \leq |\sigma|$, $\sigma(i)$ represents the i th event of σ . $L \in \mathbf{B}(A^*)$ is an event log.

An event log can be considered as a multiset of traces [1] and each trace describes the life-cycle of a particular instance (or case).

Generally speaking, a process discovery approach is able to convert an event log to a process model. This paper uses *labeled Petri net* to represent a process model, and its definition is given following [3, 9, 11–13, 18, 19].

Definition 2 (Labeled Petri net). A Labeled Petri net is a 4-tuple $PN = (P, T, F, l)$, satisfying: (1) P is a finite set of places and T is a finite set of transitions where $P \cap T = \emptyset$, $P \cup T \neq \emptyset$; (2) $F \subseteq (P \times T) \cup (T \times P)$ is set of directed arcs, called flow relation; and (3) $l \in T \rightarrow \mathcal{A}$ is a labeling function where \mathcal{A} is a set of labels and $\tau \in \mathcal{A}$ denotes invisible label.

For each $x \in P \cup T$, the set $\bullet x = \{y | (y, x) \in F\}$ is the preset (input) of x and $x \bullet = \{y | (x, y) \in F\}$ is the postset (output) of x . To describe the semantics of a labeled Petri net, we use *markings*. A marking m of PN is a multiset of places, i.e., $m \in \mathbf{B}(P)$, indicating how many tokens each place contains. Markings are states of a net. A transition $t \in T$ is *enabled* in marking $m \in \mathbf{B}(P)$, denoted as $(PN, m)[t >$ if and only if $\forall p \in \bullet t : m(p) \geq 1$. An enabled transition t may *fire* and results in a new marking m' with $m' = m - \bullet t + t \bullet$, denoted by $(PN, m)[t > (PN, m')$.

3 Problem Statement and Approach Overview

This section first explains the two research questions that we are going to address. And then, we introduce an approach overview of our solution.

3.1 Research Questions

- **RQ1:** How can we find an effective approach to get a sample log such that it is representative enough to cover all (or majority of) the behavior in the original event log?
- **RQ2:** Given a sample log, how can we measure if it is representative enough to discover a reliable process model with respect to the original event log?

The answer to the first question provides an approach to sample a large-scale event log to a relatively small one which can be used for efficient discovery. And the answer to the second one is used to evaluate the quality of the sample log with respect to the original event log. Answers to these two consecutive questions perfectly summarize the main contributions of the paper.

3.2 An Approach Overview

Figure 1 shows an overview of our approach which contains the following two parts:

- **Event Log Sampling.** By taking an event log as input, we first propose a *LogRank*-based approach to obtain its corresponding sample log with a certain ratio. Essentially, the sample log is a sub-set of the original log.
- **Quality Measure of the Sample Log.** We propose an approach to quantify the quality of the sample log. To this end, we first discover a process model by taking the sample log as an input. Note that the discovered model should guarantee 100% fitness against the sample log. Then, quality of the sample log is measured based on the fitness of the original log and the discovered model.

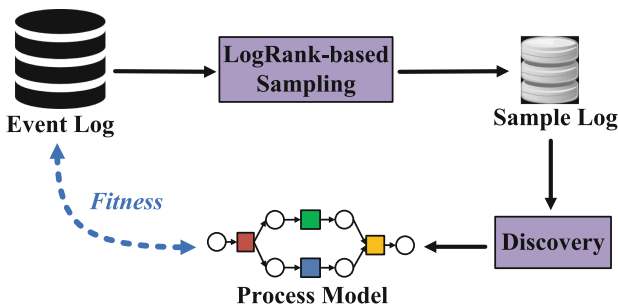


Fig. 1. An approach overview.

4 LogRank-Based Event Log Sampling

In this section, we first give an example event log, and then detail the main approach to sample an event log and quantify the quality of the sample log.

4.1 An Example Event Log

In this subsection, we give an example log (denoted as L_C) which will be used to introduce the sampling approach and the quality measure in the following subsections. This log contains 78 traces with 21 variants and 495 events in total.

$$\begin{aligned}
 L_C = [& \langle a, c, d, e, h \rangle^{16}, \langle a, b, d, e, g \rangle^9, \langle a, d, c, e, h \rangle^8, \langle a, b, d, e, h \rangle^8, \langle a, c, d, e, g \rangle^4, \\
 & \langle a, d, c, e, g \rangle^4, \langle a, d, b, e, h \rangle^4, \langle a, d, b, e, g \rangle^3, \langle a, c, d, e, f, d, b, e, h \rangle^4, \\
 & \langle a, c, d, e, f, b, d, e, h \rangle^2, \langle a, c, d, e, f, b, d, e, g \rangle^1, \langle a, c, d, e, f, d, b, e, g \rangle^1, \\
 & \langle a, d, c, e, f, c, d, e, h \rangle^1, \langle a, d, c, e, f, d, b, e, h \rangle^1, \langle a, d, c, e, f, b, d, e, g \rangle^1, \\
 & \langle a, c, d, e, f, b, d, e, f, d, b, e, g \rangle^1, \langle a, d, c, e, f, b, d, e, f, c, d, e, f, b, d, e, g \rangle^1, \\
 & \langle a, d, c, e, f, b, d, e, f, b, d, e, g \rangle^1, \langle a, d, c, e, f, d, b, e, f, b, d, e, h \rangle^1, \\
 & \langle a, d, b, e, f, b, d, e, f, d, b, e, g \rangle^1, \langle a, d, c, e, f, d, b, e, f, c, d, e, f, d, b, e, g \rangle^1, \\
 & \langle a, d, c, e, f, d, b, e, g \rangle^1, \langle a, d, c, e, f, d, b, e, f, c, d, e, f, b, d, e, f, b, d, e, g \rangle^1].
 \end{aligned}$$

4.2 A LogRank-Based Sampling Approach

PageRank is one of the most well-known methods for Google to rank their search results [15]. This algorithm organizes input as a graph and computes the scores of vertices in the graph by making use of the voting or recommendations between nodes. It has been employed in a variety of studies for importance ranking and summarization, e.g., document summarization [14, 16]. Given a graph $G = \{S, E, W\}$, where S is a set of vertices, E is a set of edges linking these vertices and W is the weight set affiliated with each edge, *PageRank* works as follows:

First, we need to transform edge weight set W into a transition matrix M where $M_{ij} = W_{(i,j)}$ (the weight on edge (i, j)) if edge $(i, j) \in E$ otherwise $M_{ij} = 0$. Second, to guarantee the transition matrix M to be a Markov transition matrix, we normalize it to make the sum of each row to be 1:

$$\tilde{M}_{ij} = \begin{cases} M_{ij} / \sum_{j=1}^{|S|} M_{ij}, & \text{if } \sum_{j=1}^{|S|} M_{ij} \neq 0 \\ 1/|S|, & \text{otherwise.} \end{cases} \quad (1)$$

Then, the *PageRank* score p_i for vertex s_i is calculated based on the transition matrix:

$$p_i = \lambda \sum_{j:j \neq i} p_j \tilde{M}_{ji} + \frac{1-\lambda}{S}. \quad (2)$$

For convenience, Eq. (2) can be denoted in the matrix form:

$$\mathbf{p} = \lambda \tilde{M}^T \mathbf{p} + \frac{1 - \lambda}{S} \mathbf{1}, \tag{3}$$

where \mathbf{p} is a $|S| \times 1$ vector that is made up of the scores of vertices in the vertex set S and $\mathbf{1}$ is a column vector with all the elements equal to 1. λ is a damping factor which ranges from 0 to 1, and $(1 - \lambda)$ indicates the probability for vertex s_i to jump to a random vertex in the graph.

Motivated by the classic *PageRank*, we propose the *LogRank* which is specially designed to order the traces and extract a summary of an event log by selecting the most representative ones. To use *LogRank* for log sampling, we treat the original log as a graph in which a trace acts as a vertex. And by linking a pair of traces in the graph, the similarity value between them is calculated as the edge weight. Specifically, we first convert a trace to a vector by selecting a typical set of features, and then compute the similarity of two vectors using the *Euclidean* distance measure [6].

Inspired by [17], we characterize traces by *profiles*, where a profile is a set of related features which describe the trace from a specific perspective. Every feature is a metric, which assigns a specific numeric value to each trace. In this way, we consider a profile with n features to be a function that maps a trace to a n -dimensional vector. In this paper, we use two types of profiles for mapping. The *activity profile* defines one feature per event name, and the *directly follow profile* defines one feature for each directly follow relation. The directly follow profile of a trace $\sigma \in L$ is defined as $dfgProfile(\sigma) = \{(\sigma(i), \sigma(i + 1)) | 1 \leq i \leq |\sigma| - 1\}$.

Table 1. Profiles of two example traces

Trace	Activity set						Directly follow relation set							
	a	b	c	d	e	g	h	(a, b)	(a, c)	(c, d)	(b, d)	(d, e)	(e, h)	(e.g.)
acdeh	1	0	1	1	1	0	1	0	1	1	0	1	1	0
abdeg	1	1	0	1	1	1	0	1	0	0	1	1	0	1

Table 1 shows the results of profiling two example traces with *activity profile* and *directly follow profile*. The profiles can be represented as a n -dimensional vector where n means the number of features extracted based on the selected profiles. Therefore, trace $\sigma_p \in L$ corresponds to vector $v_p = \langle i_{p1}, i_{p2}, \dots, i_{pn} \rangle$ such that i_{pn} denotes the existence of feature n in trace σ_p . To calculate the distance between traces, e.g., $\sigma_p, \sigma_q \in L$, we use *Euclidean* distance which can be computed as follows:

$$Distance(\sigma_p, \sigma_q) = \sqrt{\sum_{l=1}^n |i_{pl} - i_{ql}|^2} \tag{4}$$

Therefore, the similarity between σ_p and σ_q is computed as follows:

$$\text{Similarity}(\sigma_p, \sigma_q) = 1 - \text{Distance}(\sigma_p, \sigma_q) \quad (5)$$

To sum up, given an event log and a sample ratio, the procedure of *LogRank*-based sampling is described as follows:

- (1) map the log to a graph by calculating trace similarity based on Eq. (5);
- (2) calculate the PageRank value for each trace based on Eq. (3);
- (3) rank traces based on their PageRank values; and
- (4) select the top N traces according to the input sample ratio.

To illustrate the applicability of the *LogRank*-based approach, we use it to produce a set of sample logs with different ratios (from 5% to 30% with an increment of 5%) for the example log L_C in Subsect. 4.1. The basic information of the sampled logs is shown in Table 2.

Table 2. Statistics of different samples of L_C

Log name	Number of traces	Number of events	Number of event classes
Example log	78	495	8
30% Sample log	22	210	8
25% Sample log	19	195	8
20% Sample log	15	175	8
15% Sample log	12	152	8
10% Sample log	8	112	8
5% Sample log	4	68	7

4.3 Quality Measure of the Sample Log

Through *LogRank*-based sampling, we get a subset of the original log as the sample log. The sample log is typically not complete and may lead to overfitting (or underfitting) models for discovery. The goal of sampling is to improve the efficiency of process discovery without sacrificing (too much of) the quality of the model. Given a sample log, the question is if it is representative enough to discover a process model of high quality compared with that discovered directly from the original log. To this end, we propose to measure the *fitness* of the process model discovered from the sample log against the original log. According to *Buijs et al.* [2], *fitness* quantifies the extent to which a process model can accurately reproduce the traces recorded in the log. The rationale behind is that if a model discovered from the sample log can replay all (or majority of) the traces in the original log, we argue that the sample log is of high quality for process discovery.

One of the most important factors to ensure the applicability of this quality measure is that we should guarantee the model discovered from the sample log

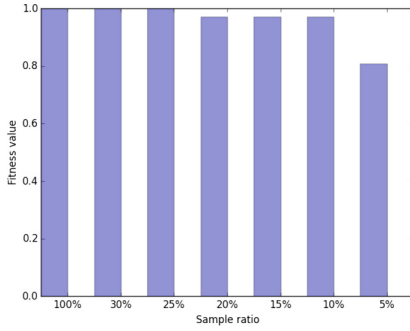


Fig. 2. Fitness comparison.

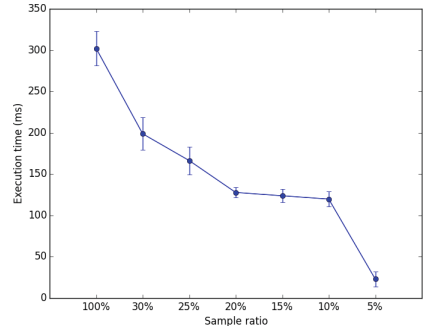


Fig. 3. Execution time comparison.

can fully represent the behavior in the sample log, i.e., 100% fitness. The rationale behind lies in that it does not make any sense to replay the original log against a model discovered from a sample log if this model cannot cover all possible behavior in the sample log. Therefore, we should select a process discovery approach that can guarantee 100% fitness, i.e., ensuring that the process model can reproduce all traces in the sample log. To our best knowledge, *Inductive Miner (IM)* [8] is one typical approach that can guarantee 100% fitness of the discovered model against the input log. Therefore, *IM* is selected in this paper.

Figure 2 shows the fitness comparison results of the process models discovered using *IM* by taking the sample logs with different ratios. According to Fig. 2, we can see that: (1) the fitness of ratios 25% and 30% are 1, i.e., the sample logs with ratios of 25% and 30% cover all information in the original log from the process discovery perspective; and (2) the fitness value decreases as the ratio decreases. This can be explained by the fact that the less information a sample log contains, the lower quality the discovered model is. To measure the discovery performance, the execution time for the original log and all sample logs is shown in Fig. 3. Generally, the execution time decreases dramatically as the size of the log decreases. Compared with the execution time of the original log, the sample log with ratio of 25% improves the discovery performance by about one time without sacrificing any quality of the discovered model. Differently, the sample log with ratio of 5% increases the discovery performance by around 8 times but losing 20% of the behavior in the original log. In summary, the sampling approach provides a solution to improve discovery efficiency, and one should be careful to find a balance between the discovery efficiency and the quality of the sample log according to real-life requirements.

5 Tool Support and Experiments

5.1 Tool Support

The open-source (Pro)cess (M)ining framework *ProM*¹ has been developed as a pluggable environment for process mining and related topics. The proposed

¹ <http://www.promtools.org/doku.php>.

LogRank-based sampling approach has been implemented as a plug-in (called *Sampling Business Process Event Log*)² in the framework. It takes an event log and a sample ratio as inputs, and returns a representative sample log as output.

5.2 Experiment Results

In this section, we perform the experimental evaluation of the *LogRank*-based sampling methodology on four different data sets (one synthetic and three real-life ones). Table 3 reports some descriptive statistics of these data sets.

Table 3. An overview of the data sets

Data set	Number of traces	Number of events	Number of event classes
Synthetic log	100	2297	20
Sepsis	1050	15214	16
BPI2011	1143	150291	624
BPI2012	13087	262200	36

In the following, we present the experimental results in light of the two research questions defined in Sect. 3.

RQ1: How can we find an effective approach to get a sample log such that it is representative enough to cover all (or majority of) the behavior in the original event log?

To answer this question, we propose a *LogRank*-based approach to obtain its the sample log. Essentially, the sample log is a representative sub-set of the original log. We produce a set of sample logs with different ratios (from 5% to 30% with an increment of 5%) for each experiment data set using our *Sampling Business Process Event Log* plug-in.

RQ2: Given a sample log, how can we measure if it is representative enough to discover a reliable process model with respect to the original event log?

Towards this question, we propose to measure the *fitness* of the process model discovered from the sample log against the original log. The rationale behind is that if a model discovered from the sample log can replay all (or majority of) the traces in the original log, we argue that the sample log is of high quality from a discovery perspective. For each sample log, we first discover a process model using the *Inductive Miner* and then replay its original log against this model to measure the quality. Detailed quality measure results (in terms of fitness) are shown in Fig. 4. In addition, the execution time of the *Inductive Miner* by taking sample logs of different ratios as inputs are also recorded and demonstrated in Fig. 5 to compare the discovery performance.

² <https://svn.win.tue.nl/repos/prom/Packages/CongLiu/>.

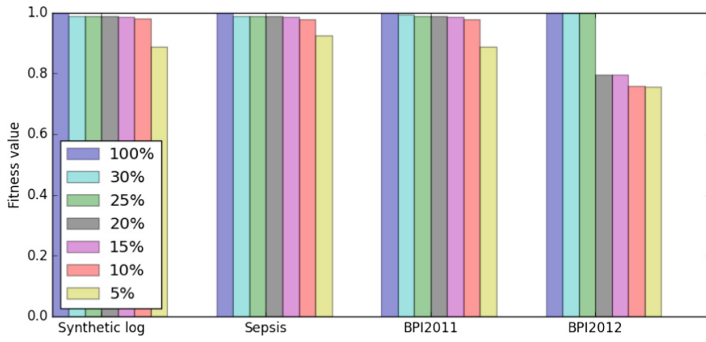
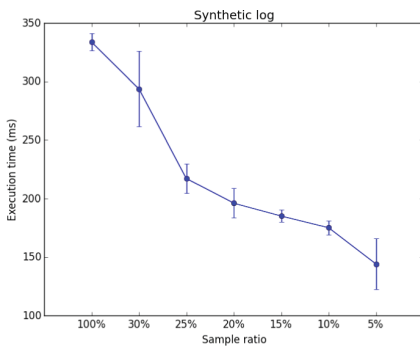
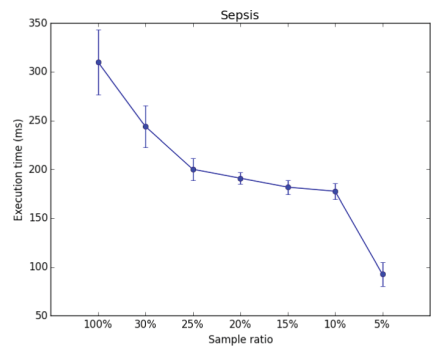


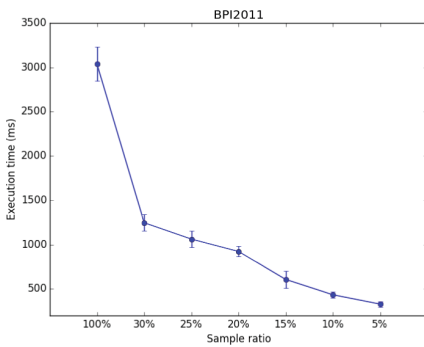
Fig. 4. Fitness comparison results.



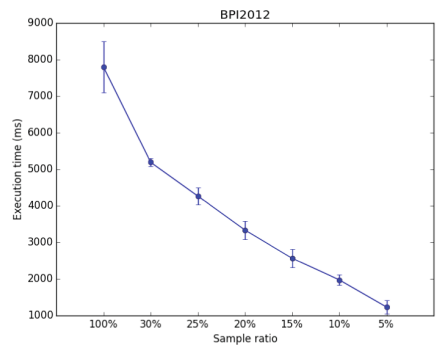
(a) Synthetic Log



(b) Sepsis Log



(c) BPI2011 Log



(d) BPI2012 Log

Fig. 5. Execution time comparison of different data set.

In general, both the execution time and the quality decrease as the size of the (sample) log decreases according to Figs. 4 and 5. Differently, the execution time reduces sharply as the size of the log decreases while the quality decrease

is relatively slow and can be maintained above a proper threshold. Considering the Sepsis data set as an example, compared with its original log the discovery performance of the sample log with 5% ratio is increased by almost 2 times but the quality decreases only from 1 to 0.889. For the BPI2012 data set, compared with its original log the discovery time of the sample log with 25% ratio is reduced from about 8000 milliseconds to 4000 milliseconds while their quality are identical (both can produce 100% fitting models). Therefore, we conclude that the *LogRank*-based sampling approach provides an effective solution to improve discovery efficiency while ensuring high quality of the discovered model, and one should be careful to find a balance between the discovery efficiency and the quality of the sample log according to the specific requirements.

6 Conclusion

Given a large-scale event log, the *LogRank*-based approach obtains a sample log using graph-based ranking model. The idea is to select a representative subset of traces from the original log. In addition, we also introduce an approach to measure the quality of a sample log with respect to the original one. The approach has been implemented as a plug-in in the ProM framework. Experimental results on four event logs show that: (1) the sampling approach provides an effective solution to improve discovery efficiency; and (2) one should find a balance between the sampling ratio (i.e., discovery efficiency) and the quality of the sample log based on the specific requirements.

This work opens the door for the following directions: (1) to give a comprehensive quality measure of the sample log, other metrics, e.g., *precision* and *generalization* [2], can be introduced; (2) to deploy our *LogRank*-based sampling approach on distributed systems [4, 5]; and (3) to explore the applicability of our *LogRank*-based sampling approach to other real-life event logs with dedicated domain knowledge is highly desired in the future.

Acknowledgement. This work was supported in part by the NSFC under Grant 61472229, Grant 61602279, Grant 71704096, and Grant 31671588, in part by the Science and Technology Development Fund of Shandong Province of China under Grant 2016ZDJS02A11, Grant 2014GGX101035, and Grant ZR2017MF027, in part by the Taishan Scholar Climbing Program of Shandong Province, and in part by the SDUST Research Fund under Grant 2015TDJH102.

References

1. van der Aalst, W.: *Process Mining: Data Science in Action*. Springer, Heidelberg (2016). <https://doi.org/10.1007/978-3-662-49851-4>
2. Buijs, J.C.A.M., van Dongen, B.F., van der Aalst, W.M.P.: On the role of fitness, precision, generalization and simplicity in process discovery. In: Meersman, R., et al. (eds.) OTM 2012. LNCS, vol. 7565, pp. 305–322. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33606-5_19

3. Cheng, J., Liu, C., Zhou, M., Zeng, Q., Ylä-Jääski, A.: Automatic composition of semantic web services based on fuzzy predicate petri nets. *IEEE Trans. Autom. Sci. Eng.* **12**(2), 680–689 (2015)
4. Cheng, L., Kotoulas, S., Ward, T.E., Theodoropoulos, G.: Robust and efficient large-large table outer joins on distributed infrastructures. In: Silva, F., Dutra, I., Santos Costa, V. (eds.) *Euro-Par 2014*. LNCS, vol. 8632, pp. 258–269. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-09873-9_22
5. Cheng, L., Li, T.: Efficient data redistribution to speedup big data analytics in large systems. In: *2016 IEEE 23rd International Conference on High Performance Computing (HiPC)*, pp. 91–100. IEEE (2016)
6. Duda, R.O., Hart, P.E., Stork, D.G.: *Pattern Classification*. Wiley, Hoboken (2012)
7. Evermann, J.: Scalable process discovery using map-reduce. *IEEE Trans. Serv. Comput.* **9**(3), 469–481 (2016)
8. Leemans, S.J.J., Fahland, D., van der Aalst, W.M.P.: Discovering block-structured process models from event logs - a constructive approach. In: Colom, J.-M., Desel, J. (eds.) *PETRI NETS 2013*. LNCS, vol. 7927, pp. 311–329. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-38697-8_17
9. Liu, C., Cheng, J., Wang, Y., Gao, S.: Time performance optimization and resource conflicts resolution for multiple project management. *IEICE Trans. Inf. Syst.* **99**(3), 650–660 (2016)
10. Liu, C., Duan, H., Qingtian, Z., Zhou, M., Lu, F., Cheng, J.: Towards comprehensive support for privacy preservation cross-organization business process mining. *IEEE Trans. Serv. Comput.* 1–15 (2016). <https://doi.org/10.1109/TSC.2016.2617331>
11. Liu, C., Zeng, Q., Duan, H., Zhou, M., Lu, F., Cheng, J.: E-net modeling and analysis of emergency response processes constrained by resources and uncertain durations. *IEEE Trans. Syst. Man Cybern.: Syst.* **45**(1), 84–96 (2015)
12. Liu, C., Zeng, Q., Zou, J., Lu, F., Wu, Q.: Invariant decomposition conditions for petri nets based on the index of transitions. *Inf. Technol. J.* **11**(7), 768–774 (2012)
13. Liu, C., Zhang, F.: Petri net based modeling and correctness verification of collaborative emergency response processes. *Cybern. Inf. Technol.* **16**(3), 122–136 (2016)
14. Mihalcea, R., Tarau, P.: *TextRank: bringing order into texts*. Association for Computational Linguistics (2004)
15. Page, L., Brin, S., Motwani, R., Winograd, T.: *The PageRank citation ranking: bringing order to the web*. Technical report, Stanford InfoLab (1999)
16. Pei, Y., Yin, W., Huang, L.: Generic multi-document summarization using topic-oriented information. In: Anthony, P., Ishizuka, M., Lukose, D. (eds.) *PRICAI 2012*. LNCS (LNAI), vol. 7458, pp. 435–446. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-32695-0_39
17. Song, M., Günther, C.W., van der Aalst, W.M.P.: Trace clustering in process mining. In: Ardagna, D., Mecella, M., Yang, J. (eds.) *BPM 2008*. LNBIP, vol. 17, pp. 109–120. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-00328-8_11
18. Zeng, Q., Liu, C., Duan, H.: Resource conflict detection and removal strategy for nondeterministic emergency response processes using petri nets. *Enterp. Inf. Syst.* **10**(7), 729–750 (2016)
19. Zeng, Q., Lu, F., Liu, C., Duan, H., Zhou, C.: Modeling and verification for cross-department collaborative business processes using extended petri nets. *IEEE Trans. Syst. Man Cybern.: Syst.* **45**(2), 349–362 (2015)
20. Zeng, Q., Sun, S.X., Duan, H., Liu, C., Wang, H.: Cross-organizational collaborative workflow mining from a multi-source log. *Decis. Support Syst.* **54**(3), 1280–1301 (2013)



Case-Based Decision Support System with Contextual Bandits Learning for Similarity Retrieval Model Selection

Booma Devi Sekar^(✉) and Hui Wang

School of Computing, Ulster University, Jordanstown Campus,
Newtownabbey, Northern Ireland, UK
{b.sekar, h.wang}@ulster.ac.uk

Abstract. Case-based reasoning has become one of the well-sought approaches that supports the development of personalized medicine. It trains on previous experience in form of resolved cases to provide solution to a new problem. In developing a case-based decision support system using case-based reasoning methodology, it is critical to have a good similarity retrieval model to retrieve the most similar cases to the query case. Various factors, including feature selection and weighting, similarity functions, case representation and knowledge model need to be considered in developing a similarity retrieval model. It is difficult to build a single most reliable similarity retrieval model, as this may differ according to the context of the user, demographic and query case. To address such challenge, the present work presents a case-based decision support system with multi-similarity retrieval models and propose contextual bandits learning algorithm to dynamically choose the most appropriate similarity retrieval model based on the context of the user, query patient and demographic data. The proposed framework is designed for DESIREE project, whose goal is to develop a web-based software ecosystem for the multidisciplinary management of primary breast cancer.

Keywords: Case-based reasoning · Clinical decision support system
Similarity retrieval · Contextual bandits learning

1 Introduction

Recent advances in healthcare industry show that there is a growing demand for personalized medicine, which aims to customize treatment to an individual patient based on his/her likelihood of response to the therapy. The move towards personalized medicine is supported by various technological advancements, especially in the area of data science, machine learning and artificial intelligence [1]. One such pathway is the development of personalized diagnostic model based on patient similarity. Case-based Reasoning (CBR), an artificial intelligent approach is very close to human reasoning, and has become a well-adapted methodology in medicine for developing personalized diagnostic model based on patient similarity measure [2]. CBR methodology adapts instance based learning, which aims to learn and derive insights from patients similar to the query patient and then analyze the derived insights in the diagnostic model to

provide personalized diagnostic/treatment recommendations to the query patient. In this paper, we present a Case-based Decision Support System (CB-DSS) for DESIREE¹, which is a European Union funded project focusing on developing a web-based software ecosystem for the personalized, collaborative, and multidisciplinary management of primary breast cancer (PBC), from diagnosis to therapy and follow-ups.

The main difference between a case-based and a rule-based system is that, the knowledge base of a case-based system is populated with cases that incorporates experts experience rather than rules defined using clinical guidelines. Secondly, in a rule-based system it is difficult to pre-define rules that explicitly match any problem, and therefore often fails to solve some of the complex problems. In a case-based approach however, a partial matching is built within the system, which allows it to provide an approximate solution to a problem. Advantage of this is that a CBR system could provide a solution to any given problem, but the challenges allies in building a CBR system that could provide a more reliable solution. This in fact, mainly relies on the first step of the CBR cycle (retrieve, reuse, revise and retain), the similarity retrieval. Building a good similarity retrieval algorithm involves various factors, from assigning proper weights to the description variables, adapting the appropriate similarity function (e.g. cosine, Euclidean, distance correlation) and incorporating general domain knowledge using ontology. Depending on different combination of these factors, the similarity retrieval algorithm could retrieve completely different cases from the case-base and thus provide varying solution to the same problem. Thus, the main challenge of building a CBR system is that there are various uncertainties involved to develop a more reliable model.

In this paper, to address the above challenge, we first present a case-based decision support system (CB-DSS) framework with multi-similarity retrieval models and propose contextual bandits learning algorithm [3] to dynamically choose the appropriate model relevant to the context of the user, query patient and demographic setting. In the following section, we first present the framework and workflow of the proposed CB-DSS. Then, we present the contextual bandit learning methodology and it's adaptation in the framework to dynamically choose between different similarity retrieval model based on the context data. Finally, we draw the conclusion.

2 Case-Based Decision Support System with Contextual Bandit Learning

2.1 Framework and Workflow

One of the main objectives of DESIREE project is to provide decision support for diversity of therapeutic options in BUs, including surgical, radiotherapy, adjuvant systemic therapies etc. With the aim to provide personalized state-of-the-art clinical decision support system to BUs, the project aims at providing guideline-DSS (GL-DSS) [4], experience-DSS (EX-DSS) [5] and CB-DSS. In this paper, we present the

¹ <http://www.desiree-project.eu>.

proposed CB-DSS using CBR methodology and contextual bandits learning algorithm. Figure 1 shows the framework and workflow of the proposed CB-DSS.

In order to incorporate decisional criteria beyond the limitations of current guidelines from breast cancer management, the CB-DSS incorporates the experience of clinicians on previous cases, by collecting description of patients, and the decision made by the clinicians, as the case representation in the data model. Also, to incorporate the knowledge from explicit domain (breast cancer), the data acquired from the clinical partners, clinical practice guidelines and clinical documentation are represented as breast cancer knowledge model (BCKM) in a Web Ontology Language (OWL), which can then be applied in similarity retrieval model using semantic similarity functions. Finally, with the feature selection made, feature weighting matrix is defined, which is also applied in the similarity retrieval model.

Next, as shown in Fig. 1, it provides a tool for querying former patient cases using similarity retrieval model. As briefed above, various factors are involved in the design of the similarity retrieval model, from feature selection and weighting, case representation, and similarity function matrix. Thus, combination of these different factors could lead one to build a completely different similarity retrieval model. For example, for a surgeon certain clinical attributes are more important than for radiologist or a general physician, therefore when defining the feature weighting matrix, one has to take into consideration the context of the user. Likewise, an oncologist will consider various non-clinical attributes of the patient such as the race, family history, and insurance status in recommending a treatment plan. The context of the patient plays an important role in making a decision. Thus, it is critical to consider the contextual information in building a similarity retrieval model.

To address the above challenges, the proposed CB-DSS framework shown in Fig. 1 is built with N number of similarity retrieval models and contextual bandit learning is proposed to exploit the context data of the user, patient and demographic data to dynamically choose the most appropriate similarity retrieval model.

Now, during runtime, the user first enters his/her details, demographic data and query patient case to the CB-DSS. The context extraction module, aggregates the context data, such as clinician's practice data, demographic data, patient family history, race etc., which are then used by the contextual bandits learning algorithm. Meanwhile, the query patient data also enters the different similarity retrieval models present in the CB-DSS. Based on the defined similarity functions and weight matrix, the similarity retrieval model compares the query case with the patient cases present in the data model to retrieve similar patient cases. The contextual bandits learning algorithm will make the decision on which similarity retrieval model will be executed by the DSS. Next section, will discuss the details on how the contextual bandits learning algorithm selects the best performing similarity retrieval model based on the contextual information.

2.2 Contextual Bandit Learning

Determining the best similarity retrieval model can be viewed as a multi-armed bandit (MAB) problem [6], where the clinician has to choose amongst a set of available arms (retrieved patient cases) and he/she can only receive the reward (see the outcome) of the

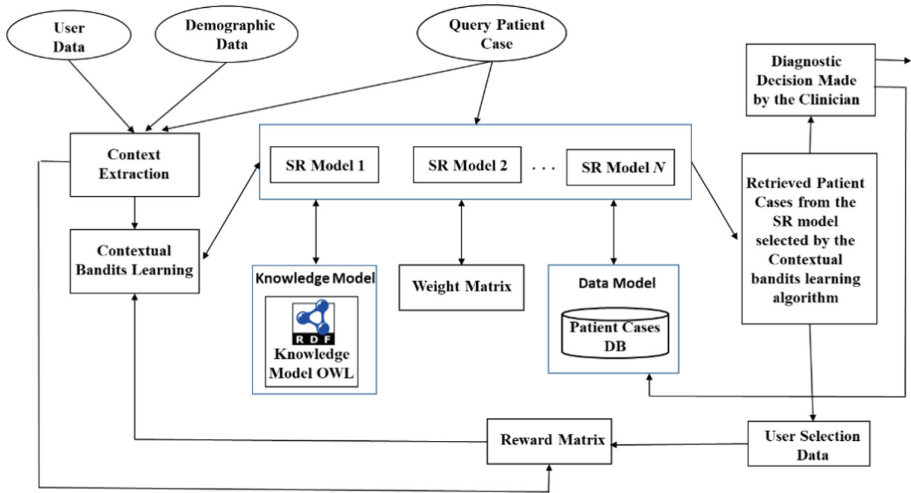


Fig. 1. Framework and workflow of CB-DSS with contextual bandits learning

action (diagnostic decision) that was taken. The clinician will not be able to know the possible outcome, if his/her decision was based on the choice of the patient case retrieved from a different case retrieval model.

To solve the above problem, the proposed algorithm should learn to choose the actions that can maximize the rewards (choose the decision from best performing model). A contextual bandit learning algorithm addresses such problem by providing the context (a hint about the reward) before the action is actually taken. The context in our model that can determine the reward can be derived from various factors, such as the demographic data (breast unit in the hospital), user (e.g. radiologist, oncologist or general physician), and query patient (physiological data, such as race).

Now with the above information, a contextual bandit framework can be defined as follows. Let X be the context set and A be the arms set (action). In each round of the algorithm t , $t = 1, \dots, T$, T is the time zone, the following events is executed in succession:

1. a context $x_t \in X$ is observed by the learner,
2. based on the observed context, a reward vector $r_t \in [0, 1]^K$ is chosen, but not received by the learner,
3. learner chooses an arm (action) $a_t \in \{1, \dots, K\}$,
4. learner receives the reward $r_t(a_t)$.

Now, the goal of a bandit algorithm is to maximize the total reward $\sum_{t=1}^T r_t(a_t)$. So, in order to maximize the reward, the algorithm should execute a good policy π (e.g. decision rule) to allow the learner to choose an action based on the context. The algorithm will have to work in a rich policy space $\Pi = \{\pi : X \rightarrow A\}$ that could be extremely large. Thus, it has to efficiently learn about all policies and choose the best policy. Therefore, when the arm is selected, the learner will observe reward for policies

that would have chosen the same arm. Now, the aim is to obtain a high total reward relative to the best policy $\in \Pi$, computed as minimum contextual regret C_r as shown in Eq. (1). Where the first term in Eq. (1) is the average reward for the best policy and the second term is the learner’s average reward.

$$C_r = \max_{\pi \in \Pi} \frac{1}{T} \sum_{t=1}^T r_t(\pi(x_t)) - \frac{1}{T} \sum_{t=1}^T r_t(a_t) \quad (1)$$

The goal of the above Eq. (1) is to bring the C_r quickly to zero. Various contextual bandits learning algorithm, including ϵ -greedy, ϵ -first, ϵ -decreasing, contextual ϵ -greedy, bagging, upper confidence bound, lower confidence bound, Thompson sampling, and bandit forest are present in literature [7–9]. Among which, ϵ -greedy is the most fastest and simplest approach that can be adapted, which exploits the best strategy with probability of $(1 - \epsilon)$ and uniformly exploits over all the other actions with probability of (ϵ) . The regret computed with ϵ -greedy algorithm is shown in Eq. (2).

$$r_t = O\left(\left(\frac{K \ln |\Pi|}{T}\right)^{1/3}\right) \quad (2)$$

As the regret is to the power of 1/3, it may not be the most optimal bandits learning algorithm. However, it is computationally efficient, when working with a larger data set. Thus, as the next step ϵ -greedy algorithm will be applied as the contextual bandits learning algorithm in the proposed framework of CB-DSS to optimize the selection of the similarity retrieval model.

2.3 A Running Example

In this section, with an example, we will demonstrate on how contextual bandits learning can help in identifying the optimal similarity retrieval model in the CB-DSS for breast cancer management.

The main goal of contextual bandits learning algorithm is to maximize the total reward achieved by the learner, i.e. obtain the minimum contextual regret. As there exists, a policy π (decisional rule) that can give high rewards, the contextual bandits learning algorithm has to efficiently learn from all policies and choose the best policy. In our example let’s assume ‘ n ’ number of policies are defined using decisional rules (IF-THEN statements) for different contexts. For example, “IF Surgeon THEN SR Model 1”, “IF Radiologist THEN SR Model 2”, where each SR model is assigned with a different weight matrix and similarity function model.

Now, during run time, the query case is sent to the SR model to retrieve similar cases from the patient case-base. Simultaneously, the context of the user, demographic information and patient data are sent to the contextual learning algorithm to enable it to select an optimal SR model. The selected SR model will now retrieve 10 similar cases to the query case. Based on the patient case selected by the user to make the clinical decision, the learner receives the corresponding reward. As shown in Table 1, we assign 1.0 for the most similar patient case and 0.1 to the 10th similar case.

Table 1. Reward value assigned for the learner’s action

Context	Action (Retrieved Patient Cases)								
	1	2	3	4	...	8	9	10	
GP, Outpatient, Asian	1.0	0.9	<u>0.8</u>	0.7	...	0.3	0.2	0.1	
Radiologist, Radiology, European	1.0	0.9	0.8	0.7	...	0.3	0.2	0.1	
Oncologist, Oncology, African	1.0	0.9	0.8	<u>0.7</u>	...	0.3	0.2	0.1	
...	...								

The reward received by the learner is marked in **Bold**.

The reward received by the best policy is Underlined.

In the Bandits setting, as the learner could only observe the reward for the action taken, from Table 1, the learner’s total reward can be computed as ‘0.9 + 0.3 + 0.2 + ...’. Meanwhile, as the learner and the best policy have chosen the same arm for the second user, only the policy’s reward of 0.3 is known. Here the best policy is determined to be the one, which would have chosen the same case with possibly a higher reward.

The contextual bandits learning algorithm is applied to exploit and explore, i.e. exploit the information available and explore from the action taken to learn and choose the best policy that gives the minimum regret and therefore the optimal result. In ϵ -greedy contextual bandits learning algorithm, it exploits the best strategy with probability of $(1 - \epsilon)$ and uniformly exploits over all the other actions with probability of (ϵ) , until optimal solution is achieved.

3 Conclusion

In this work, we have developed a CB-DSS for DESIREE project, aimed at providing web-based software for breast cancer diagnosis and management. The proposed CB-DSS provides a tool for querying former cases in order to retrieve similar patient cases from the case-base. As we note that the design of similarity retrieval model involves various factors from feature selection and weighing, similarity function, case representation and knowledge model, developing an optimal similarity retrieval model is challenging. To address such challenge, we presented a CB-DSS framework with multi-similarity retrieval models. We propose contextual bandits learning algorithm to dynamically choose between different similarity retrieval models by learning from the contextual information extracted from the user, patient and demographic data. The paper presents the overall framework of the proposed CB-DSS and systematically describes its workflow with a running example.



Acknowledgments. The DESIREE project has received funding from the European Union’s Horizon 2020 research and innovation program under grant agreement No. 690238.

References

1. Parra-Calderón, C.L.: Patient similarity in prediction models based on health data: A scoping review, *JMIR Med Inform.* **5**(1) (2017)
2. Alexandrini, F., Krechel, D., Maximini, K., Wanggenheim, A.: Integrating CBR into the health caser organization. In: 16th IEEE Symposium on Computer-Based Medical Systems (2003)
3. Mary, J., Gaudel, R., Preux, P.: Bandits and recommender systems. In: Pardalos, P., Pavone, M., Farinella, G.M., Cutello, V. (eds.) *MOD 2015*. LNCS, vol. 9432, pp. 325–336. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-27926-8_29
4. Séroussi, B., et al.: Reconciliation of multiple guidelines for decision support: a case study on the multidisciplinary management of breast cancer within the DESIREE project. In: 2017 Proceedings of the AMIA Annual Symposium, Washington DC, 4–8 November 2017 (2017)
5. Larburu, N., et al.: Augmenting guideline-based CDSS with experts' knowledge. In: 10th International Conference on Health Informatics, Porto, Portugal, 21–23 February 2017 (2017)
6. Bouneffouf, D., Feraud, R.: Multi-armed bandit problem with known trend. *Neurocomputing* **205**, 16–21 (2016)
7. Langford, J., Zhang, T.: The Epoch-Greedy algorithm for multi-armed bandits with side information. In: *Advances in Neural Information Processing System*, pp. 817–824 (2008)
8. Bouneffouf, D., Bouzeghoub, A., Gançarski, A.L.: A contextual-bandit algorithm for mobile context-aware recommender system. In: Huang, T., Zeng, Z., Li, C., Leung, C.S. (eds.) *ICONIP 2012*. LNCS, vol. 7665, pp. 324–331. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-34487-9_40
9. Agrawal, S., Goyal, N.: Thompson sampling for contextual bandits with linear payoffs. In: *ICML* (3), pp. 127–135 (2013)



Cross-Layer Attack Path Exploration for Smart Grid Based on Knowledge of Target Network

WenJie Kang¹, PeiDong Zhu², Gang Hu¹, Zhi Hang³, and Xin Liu⁴

¹ College of Computer, National University of Defense Technology, Changsha 410073, China

{kangwenjie,hugang}@nudt.edu.cn

² Department of Electronic Information and Electrical Engineering, Changsha University, Changsha 410022, China

pdzhu@nudt.edu.cn

³ Key Laboratory of Hunan Province for Mobile Business Intelligence, Changsha 410205, China

hangzhi759250163.com

⁴ Department of Computer Engineering and Applied Math, Changsha University, Changsha 410022, China

liuxinbishen@126.com

Abstract. Attack path has obviously changed due to multiple-layer structure and the characteristic of failure cross-layer propagation, which changes from static to dynamic and from single layer to multilayer. Attack path exploration is meaningful for simulating the attacker's intention and is convenient for the defenders to develop a defense mechanism. In this paper, based on a knowledge of target network (i.e., the state of cyber nodes, power flow, node type, voltage, active power, reactive power and time factor etc.), we firstly propose forward and inverse bi-directional solution model that utilizes thread propagation mechanism in the communication network and failure diffusion mechanism in power grid to explore multiple accessible cross-layer attack paths (CLAPs). Thread propagation mechanism considers system vulnerability, threat propagation, and time factor. Failure diffusion mechanism utilizes power flow to trigger load distribution in order to cause attack targets to fail. Secondly, we describe the concept of cross-layer attack path and classify it as four types: Direct Attack Path (DAP), Threat Propagation Attack Path (TPAP), Failure Diffusion Attack Path (FDAP), and Threat Propagation and Failure Diffusion Attack Path (TPFDAP). Thirdly, we propose an assessment method to evaluate the generation probability of CLAPs. Finally, experimental results show that the CLAP of the smart grid can be accurately identified in time, and the defenders can predict the best possible CLAP according to its generation probability. The CLAPs of the same targets are different at the different times and are easily affected by the state of the cyber layer and the tolerance α of the physical layer.

Supported by National Natural Science Foundation of China (Grants 61572514 and 61501482) and Changsha Science and Technology Program (Grant K1705007).

© Springer Nature Switzerland AG 2018

W. Liu et al. (Eds.): KSEM 2018, LNAI 11061, pp. 433–441, 2018.

https://doi.org/10.1007/978-3-319-99365-2_38

Keywords: Cross-layer attack path (CLAP)
Threat propagation mechanism · Failure diffusion mechanism

1 Introduction

As an important direction of network security technology, the attack path received sustained attention by experts and scholars. The related research on the attack path is mainly divided into identification, reconstruction, and exploration. Attack path identification, reconstruction, and exploration occur separately in the attack, after the attack, and before the attack. However, attack path exploration is more important to exploit the vulnerability of network defense via simulating attacker behavior and understanding attack intention. Current researchers mainly focus on attack paths of the cyber layer, and rarely integrate the characteristics of the physical layer into the generation of cross-layer attack paths. As networks of networks, the interdependence characteristics and security vulnerability may be used by the attackers to destroy the physical targets. Attack path exploration is not only conducive to seeking the more effective and low-cost attack path but also helps defenders to develop defense strategies.

Kumar et al. [1] married fault trees and attack trees to propose attack fault trees (AFTs) for identifying the most likely attack path and analyzing the expected impact of an attack. A monitoring layer attack modeling method was proposed to reconstruct attack paths in power system and was essential to the future development of defense mechanism against the attack [2]. Exploitation of vulnerabilities was often the choice of attackers for launching an attack that provides paths for gaining access to restricted resources and generates attack graph [3]. Based on automatic generation of attack path, the defense simulation model was proposed to deal with the DOS attack based on the agent technology [4]. A new nine tuples attack graph model was proposed to combine with Attack Threat Index (ATI) and the resource vulnerability index to predict attack paths [5].

The rest of the paper is organized as follows. In Sect. 2, we propose a forward and inverse bi-directional exploration mode to find cross-layer attack paths from source cyber nodes to targeted physical nodes by combining threat propagation mechanism with failure diffusion mechanism. In Sect. 3, we present the mathematic model for assessing generation probability of attack paths. Section 4 shows the experimental results. Finally, conclusions are given in Sect. 5.

2 Forward and Inverse Bi-Directional Exploration Model

The smart grid is the super-network that consists of the cyber layer and physical layer. The vulnerability in the cyber layer and the power flow in the physical layer are easy for attackers to create opportunities to form multiple cross-layer attack path. The physical layer provides power support to the cyber layer. The forward and inverse bi-directional exploration model in which forward exploration is to

find attack path from the source cyber nodes to the targeted cyber nodes and inverse exploration is to find an inverse attack path from the targeted physical nodes to the source physical nodes. In particular, the source cyber nodes is nodes with loophole or vulnerabilities and the failure of the source physical node can cause the targeted physical node to fail.

2.1 Threat Propagation Mechanism in the Communication Network

The characteristics of transmission network (e.g. connectivity or the length of the shortest path between substations) are used to specify how malicious attacks can propagate through cyber-physical systems [6]. Each cyber node has two states of insecurity and security before being attacked. Insecurity state indicates that the cyber nodes have vulnerabilities or loopholes. Security state represents that there are no loopholes in the cyber nodes. Threat Propagation Mechanism is to first attack insecurity nodes and then utilize the trust or superior-to-subordinate relationships between neighboring nodes to propagate the threat (false commands or viruses) to more distant nodes or targeted cyber nodes.

The threat propagation dynamic process that can be described by Formula 1, which means that the state of a node may change with time and depends on itself and its neighbors. $x_i(t)$ denotes the state of node i in time t and $x_i(t+1) > 0$ or $x_i(t+1) = 0$ indicate that the node state is insecurity or security state of node i at time $t+1$, respectively. α_{ii} denotes the probability of node i being attacked directly. β_{ji} denotes the probability of node i being attacked indirectly from node j .

$$x_i(t+1) = \alpha_{ii}x_i(t) + \sum_{i \in N_i} x_j(t)\beta_{ji} \quad (1)$$

2.2 Failure Diffusion Mechanism in the Power Grid

As a fundamental character of the power grid, the power flow can cause load redistribution of the branches and may lead to the failure of more nodes due to overload [7]. Failure diffusion mechanism (FDM) is to utilize load redistribution mechanism in order to trigger cascading failure for inducing targeted physical node failures. This means that a failed node can cause a new round load distribution so that the failure is diffused to more distant nodes. Since the voltage is associated with active power and reactive power, the voltage is used as the node load.

Definition 1. The capacity of the node is defined as a special ability to handle load changes [8]. α is tolerance parameter and \pm indicates that the system can withstand the range of load changes.

$$C(v_i) = (1 \pm \alpha)L(v_i) \quad (2)$$

Definition 2 (Failure Nodes Set). $S_f^\alpha(v_i)$ represents failure node set (FNS) under tolerance α after node i is removed. As α increases, the size of failure node set S_f^α will decrease. If we obtain *FNS* of all the nodes, it is easy to know that a failed node can cause which nodes to fail.

Definition 3 (Inverse Set of *FNS*). An inverse set of node j is given by $S_{f-1}^\alpha(v_j)$, in which any node being attacked will lead to the failure of node j . Special note $S_{f-1}^\alpha(v_j) = S_f^\alpha(v_j)^{-1}$. For instance, in the case of $\alpha = 0.1$, failed node 1 can cause the failure of nodes 2, 3, and 5, and failed node 2 can cause the failure of nodes 4 and 5. It is easily to know that an effective way of disabling node 5 is to disable node 1 or 2. Therefore, this can be written as follow: $S_f^{0.1}(v_1) = \{v_2, v_3, v_5\}$, $S_f^{0.1}(v_2) = \{v_4, v_5\}$ and $S_f^{0.1}(v_5) = \{v_1, v_2\}$.

2.3 Attack Targets

The criticality of substations can be used as an evaluation index to assess the impact of nodes on the power grid. The criticality of substations is described as:

$$C_i = T_i * (a'_i \frac{\sqrt{P_i^2 + Q_i^2}}{\sqrt{P_{Max}^2 + Q_{Max}^2}} + b'_i \frac{Vol_i}{Vol_{Max}} + c'_i \frac{D_i}{D_{Max}}) \tag{3}$$

where P_i , Q_i , and Vol_i represent the active power, reactive power, voltage of substation i , respectively. T_i is the type of substation i . The type value of generator, transmission and distribution station is equal to 0.95, 0.8 and 0.6, respectively. D_i is the degree of substation i . a'_i , b'_i and c'_i are equal to 0.5, 0.3 and 0.2, respectively. a'_i, b'_i and c'_i meet the condition $a'_i + b'_i + c'_i = 1$.

3 Cross-Layer Attack Path of the Smart Grid

Cross-layer attack conception is proposed to adapt to the complex and multiple-layer network environment by simulating the attackers' behavior. In Fig. 1, cross-layer attack paths (CLAPs) have four types: (I) Direct Attack Path (DAP) often occurs in the situation that the targeted physical nodes are coupled with the source cyber nodes. For instance, $c1 \rightarrow p1$ and $c6 \rightarrow p6$ are CLAPs in Fig. 1(a) and (b). (II) Threat Propagation Attack Path(TPAP) to utilize TPM for sending error control commands from the source cyber node to targeted cyber nodes, and control targeted cyber nodes to cause the targeted physical nodes to fail. For instance, $c6 \rightarrow c4 \rightarrow p4$ is generated by attacking insecurity node $c6$ in Fig. 1(c) and (d). (III) Failure Diffusion Attack Path (FDAP) is to attack the targeted cyber nodes and utilize FDM to cause the failure of the targeted physical nodes due to overload. For instance, $c1 \rightarrow p1 \rightarrow p2 \rightarrow p3 \rightarrow p4$ is formed in Fig. 1(e) and (f). (VI) Threat Propagation and Failure Diffusion Attack Path (TPFDAP) is to utilize TPM for propagating error commands from the source cyber nodes to targeted cyber nodes and utilize FDM for triggering load redistribution of source

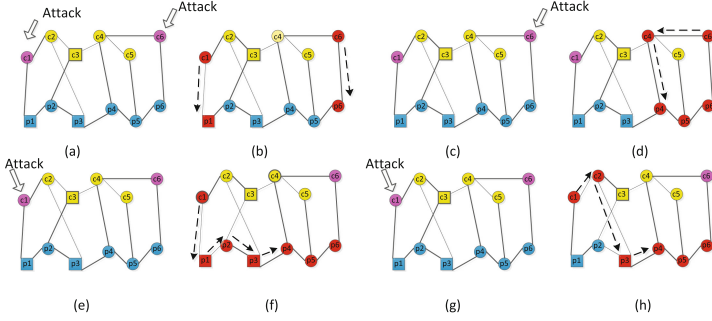


Fig. 1. The diagram of the four types of the cross-layer attack paths. The upper layer is the communication network in which square node represents the control center and circular nodes represent sensors. Yellow and blue represents secure nodes and pink represent insecurity nodes. The lower layer is the power grid in which square nodes (p1 and p3) represent generators and circular nodes represent the load nodes. Red represents the failed nodes and dotted black lines with arrows represent attack paths. (Color figure online)

physical nodes to cause the failure of targeted physical nodes. For instance, $c1 \rightarrow c2 \rightarrow p3 \rightarrow p4$ is generated from $c1$ to $c2$ and from $p3$ to $p4$ in Fig. 1(g) and (h).

Generation probability (GP) is used to evaluate the importance of CLAPs, which represent the possibility of the targeted physical nodes being successfully attacked by CLAPs. The state of cyber nodes changes dynamically with time, which means that attack paths may be different in different times. The state evolution function, threat propagation probability and tolerance are related to generation probability of CLAPs that can be described as:

$$GP_t = \prod_{i \in V_{att}} \varphi_i p_{i1} \cdots p_{kj}, \quad j \in V_{target}^P \text{ or } k \in S_{f-1}^\alpha V_{target}^P \quad (4)$$

where $p_{ij} = t_{ij} \cdot \varphi_j = \frac{2d_i \cdot d_j}{d_i^2 + d_j^2} \cdot \varphi_j, i, j \in V^C$. This means that the probability of attack path from node i to node j depends on trust degree between them and the state of node j . d_i represents the degree of node i , and $(2d_i d_j)/(d_i^2 + d_j^2)$ indicates that the two nodes with the same degree have a high level of trust. $k \in S_{f-1}^\alpha V_{target}^P$ represents that node k belongs to an inverse set of targeted physical node j in which any failed node can lead to the failure of target j .

Figure 2 shows the influence of repair time and the probability of successful defense on the evolution of the interdependent networks. φ_i represents the state evolution function that depends on repair time Δt_k and the probability ρ_i , where \bar{t}_i is the initial attacking time and \tilde{t}_i is the repairing time. $\rho_i = \frac{DC_i \cdot RA_i / AC_i}{\max(DC_i \cdot RA_i) / \min(AC_i)}$ denotes the probability of successful defense, where RA_i is resource allocation rate that defenders are supposed as be smart and allocate resources (i.e., manpower, material, and financial resources) according to the importance of node i . AC_i represents the attack capability or technolog-

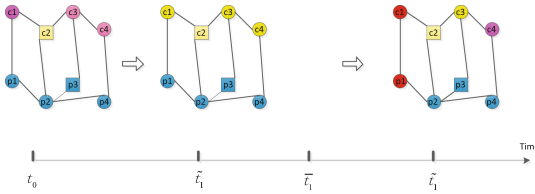


Fig. 2. The evolution of an interdependent network consisting of a communication network and a power grid in the presence. When $\Delta t_k = \tilde{t}_i - \bar{t}_i \leq 0$, $\varphi_i = 0$ means that the cyber node can respond to the change of node i and its vulnerabilities can be repaired in time to avoid being attacked. If $\Delta t_k > 0$, $\varphi_i = 1 - \rho_i^{\Delta t_k}$ means that the vulnerabilities of the cyber nodes are not repaired in time. (Color figure online)

ical level on node i , and DC_i is the defense capability or technological level on node i .

$$\varphi_i = \begin{cases} 0, & \Delta t_k \leq 0 \\ 1 - \rho_i^{\Delta t_k}, & \Delta t_k > 0 \end{cases} \quad (5)$$

4 Simulation and Experiment

In this section, the IEEE 118-bus system is used to verify our model, which contains 19 generators (red nodes), 99 transmission stations (green nodes) and 117 links. We built a communication network that contains 1 control center (red node) and 117 monitoring/controlling nodes (green nodes). According to Formula 3, physical nodes 59, 98, 85, 82, 93 and 89 can be used as targets. Figures 3 (a)-(c) show the cross-lay attack paths (CLAPs) of different states in t_1 , t_2 , and t_3 , respectively. The cyber nodes $c77$, $c87$, $c12$, $c98$, $c43$, $c75$, $c104$, $c108$, $c80$, $c30$, and $c65$ have vulnerabilities in t_1 . Similarly, the source cyber nodes are $c104$, $c118$, $c81$, $c30$, $c66$, and $c75$ at t_2 , and the source cyber nodes are $c105$, $c110$ and $c34$ at t_3 . It is clear that source cyber nodes gradually reduced due to some of them being repaired, and there are loopholes in some new nodes. Therefore, the cyber nodes have different states at different times, and different states can generate different CLAPs. Figures 3 (d)-(f) show the situation of the CLAPs that are from $c105$, $c110$ and $c34$ to $p58$ when the time equals t_3 . As tolerance α increases, the number of the CLAPs decreases. For instance, Figs. 3 (d), (e) and (f) show that the number of CLAPs is equal to 19 under $\alpha = 0.005$, 9 under $\alpha = 0.01$ and 2 under $\alpha = 0.05$, respectively. Figure 3 (e) shows that the CLAPs are $c105 \rightarrow c12 \rightarrow c56 \rightarrow p49 \rightarrow p58$, $c105 \rightarrow c12 \rightarrow c75 \rightarrow p51 \rightarrow p58$ and $c105 \rightarrow c12 \rightarrow c67 \rightarrow p58$ etc. This means that attackers not only use TPM to indirectly attack $c56$, $c75$, and $c67$ from $c105$ but can also utilize FDM to attack $p49$ and $p51$ in order to cause the failure of $p58$.

Table 1 shows the situation of CLAPs of Fig. 3 (e), which contains the mode, sequence, and generation probability of the CLAPs. According to Formula 4 and 5, we can obtain the value of generation probability of these paths, and

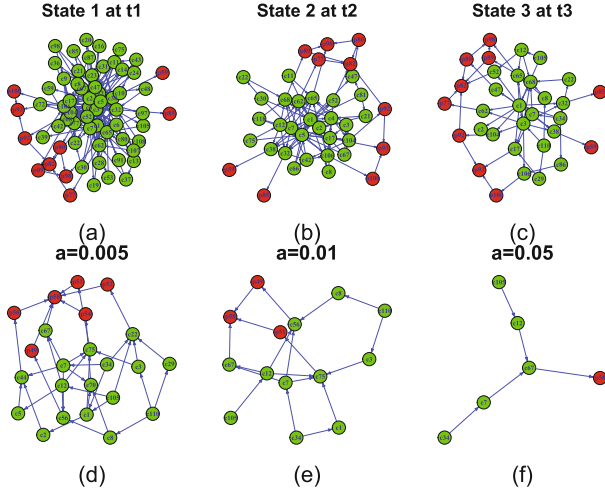


Fig. 3. The cross-layer attack paths (CLAPs) of targets p59, p98, p85, p82, p93 and p89 in the state of a communication network at (a) t1, (b) t2, and (c) t3. The CLAPs of target p58 under (d) $\alpha = 0.005$, (e) $\alpha = 0.01$, (f) $\alpha = 0.05$ at t3. Green and red nodes represent the cyber and physical nodes, respectively. Blue lines with arrows represent attack paths. (Color figure online)

then select a path with the bigger GP . Because the generation probability of $c110 \rightarrow c8 \rightarrow c56 \rightarrow p49 \rightarrow p58$ is equal to 0.23807596 and is larger than other paths, so it will be seen as the best cross-layer attack path for attackers.

we use the dataset of the real network to verify our methods, e.g., the Italian power system consists of the Italian High-Voltage (380 kV) Electrical Transmission (HVIET) network with 310 substations, 361 transmission links and the communication network with 3 control centers, 307 sensors [8]. Experimental results show our method can quickly and accurately find multiple reachable

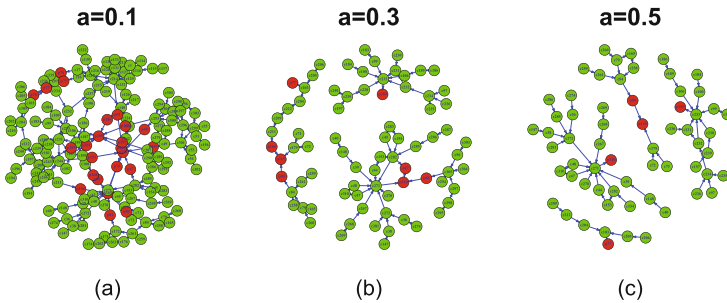


Fig. 4. The cross-layer attack paths (CLAPs) from all cyber nodes to targets p77, p165, p289, p310 in the Italian power grid under different α . (a) $\alpha = 0.1$. (b) $\alpha = 0.3$. (c) $\alpha = 0.5$.

CLAPs, for instance, when p77, p165, p289, p310 are used as targets, CLAPs can be predicted from all cyber nodes under different α as shown Figs. 4(a)-(c). The number of CLAPs under $\alpha = 0.1$, $\alpha = 0.3$, and $\alpha = 0.5$ is 131, 44, and 32, respectively. This means that the number of CLAPs in the Italian power system is negatively related to tolerance α .

Table 1. The mode, sequence and generation probability of CLAPs.

Attack path mode	Cross-layer attack path sequence	Generation probability
TPFDAP	$c105 \rightarrow c12 \rightarrow c56 \rightarrow p49 \rightarrow p58$	0.072221175
TPFDAP	$c105 \rightarrow c12 \rightarrow c75 \rightarrow p51 \rightarrow p58$	0.060591716
TPAP	$c105 \rightarrow c12 \rightarrow c67 \rightarrow p58$	0.083593786
TPFDAP	$c110 \rightarrow c8 \rightarrow c56 \rightarrow p49 \rightarrow p58$	0.23807596
TPFDAP	$c110 \rightarrow c3 \rightarrow c75 \rightarrow p51 \rightarrow p58$	0.024880836
TPFDAP	$c34 \rightarrow c7 \rightarrow c56 \rightarrow p49 \rightarrow p58$	0.05188292
TPFDAP	$c34 \rightarrow c7 \rightarrow c75 \rightarrow p51 \rightarrow p58$	0.04336696
TPFDAP	$c34 \rightarrow c1 \rightarrow c75 \rightarrow p51 \rightarrow p58$	0.041990012
TPAP	$c34 \rightarrow c7 \rightarrow c67 \rightarrow p58$	0.060314424

5 Conclusion

In this paper, we utilize TPM and FDM to explore the dynamic cross-layer attack paths. We proposed the forward and inverse bi-directional exploration model to find a low-cost and proper cross-layer attack path. Experimental results are drawn as: (I) the tolerance α is negatively correlated with the number of CLAPs; (II) the CLAPs change with the state of cyber nodes and α ; (III) the forward and inverse bi-directional exploration model can dynamically generate multiple CLAPs. In future, we can extend the model to adopt more complex cyber-physical systems and multilayer coupling networks. The inverse problem-solving mathematics method can be used to explore dynamic CLAPs.

References

- Gonda, T., Puzis, R., Shapira, B.: Scalable attack path finding for increased security. In: Dolev, S., Lodha, S. (eds.) CSCML 2017. LNCS, vol. 10332, pp. 234–249. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-60080-2_18
- Wang, J.K., Moya, C.: Attack path reconstruction from adverse consequences on power grids with a focus on monitoring-layer attacks. In: Joint Workshop on Cyber-Physical Security and Resilience in Smart Grids. IEEE Computer Society, pp. 1–6 (2016)
- Garg, U., Bansal, S., Prashar, D., et al.: Inter-dependent effect of vulnerabilities for generation of effective attack path score. Int. J. Appl. Eng. Res. **10**(8), 21487–21499 (2015)

4. Xin, J.: Defense simulation task deployment method based on attack path automatic generation. *Autom. Instrum.* (2017)
5. Wang, H., Wang, T., Liu, S.: A network attack path prediction method based on ATI. *Comput. Eng.* **42**(9), 132–137, 143 (2016)
6. Hines, P., Cotilla-Sanchez, E., Blumsack, S.: Do topological models provide good information about electricity infrastructure vulnerability? *Chaos: Interdisc. J. Non-linear Sci.* **20**, 3 (2010)
7. Liu, X., Li, Z.: Local load redistribution attacks in power systems with incomplete network information. *IEEE Trans. Smart Grid* **5**(4), 1665–1676 (2014)
8. Kang, W., Hu, G., Zhu, P., Liu, Q., Hang, Z., Liu, X.: Influence of different coupling modes on the robustness of smart grid under targeted attack. *Sensors* **18**(6), 1699 (2018)



Exploring Cyber-Security Issues in Vessel Traffic Services

Eleni Maria Kalogeraki^(✉), Spyridon Papastergiou, Nineta Polemi, Christos Douligeris, and Themis Panayiotopoulos

University of Piraeus, 80 Karaoli and Dimitriou Str., 18534 Piraeus, Greece
{elmaklg, paps, dpolemi, cdoulig, themisp}@unipi.gr

Abstract. In recent digital evolution years, cyber-terrorist activity is increasingly rising all over the world deploying new methods, using advanced technologies and sophisticated weapons. A potential terrorist attack on a large commercial Port could lead to dramatic losses. This work aims to illustrate methods for recognizing cyber-threats and security weaknesses on the ports' Critical Infrastructures and explores how these issues can be systematically exploited to harm ports and their vicinity. To this end, we follow an asset-centric approach, which employs knowledge representation techniques, to detect vulnerability chains and possible attack-paths on ports' assets. Considering the results, a realistic coordinated cyber-attack scenario on the application case of the Cruise Vessel Traffic Service is presented to show how cyber-attacks can be realized by terrorists on commercial ports.

Keywords: Attack paths · Cruise vessel traffic · Vulnerability chain

1 Introduction

Maritime installations require advanced protection mechanisms with effective risk management techniques to ensure their integrity and safety. In the past cyber-attacks were motivated to satisfy personal or financial purposes, while nowadays, the cyber attacker's profile is changing [1]. Combating terrorism is now a top priority for the European countries and their ports. Maritime terrorism is not far from becoming a reality [2]. Automatic Identification Systems (AIS), navigation services, the Electronic Chart Display and Information System (ECDIS) system and Remote Control mechanisms are cyber-assets of a high security risk because of their data interchange nature [1]. Moreover, ports are considered soft-targets because of the low security protection used for their equipment and the absence of adoption of additional encryption authentication practices [1]. The cruise service research area, has seen a variety of studies focusing on the relationship between marine-based travel and tourism, travel medicine and travelers' well being and protection on specific sensitive cases, besides, scenario planning and business analytics have not been extensively investigated [3].

Even though, terrorism in the tourism market has already attracted some attention [3]; maritime terrorism is a research area of limited academic research [4].

Concerning the maritime cybersecurity research domain, limited attention has been paid in addressing practices that could help the maritime decision makers to better

understand the cybersecurity awareness of the sector [5]. In this context, adopting Knowledge Management techniques can facilitate the expressiveness and externalization of tacit maritime cybersecurity issues and provide effective methods to create new knowledge and enhance the cybersecurity awareness [6].

The aim of this work is to present an asset-centric approach, which adopts semantic web technologies and risk management practices to recognize cybersecurity issues in the Maritime Industry and to show how vulnerability chains can be followed for attack path detection on port Critical Infrastructures, such as the Cruise Vessels Traffic Management System. In addition, a terrorist attack scenario is formed containing a series of cyber-attacks to illustrate how terrorists can take advantage of maritime cybersecurity weaknesses to launch coordinated cyber-attacks on commercial ports.

This work is based on previous dedicated research efforts [6–8] and is conducted in the realm of SAURON [9], EU H2020 project that aims to address physical and cyber threats as well as combined ones, which could potentially affect ports. Its vision is to provide a multidimensional yet installation-specific Situational Awareness solution that aims to recognize and avert cyber, physical or combined threats, to empower the safety of ports cargo business and their involved entities in the vicinity.

The rest of the paper is organized as follows: Sect. 2 describes the related research work, Sect. 3 presents the asset-centric approach, analyzed in sequential steps to identify vulnerabilities and cyber-threats and discover possible attack paths that could be used by adversaries to take advantage of port's Critical Infrastructures. Finally, in Sect. 4, we draw conclusions of our work.

2 Related Work

Maritime terrorism has been neglected in tourism research and a great number of citizens believe that the most challenging threat in ports is a terror attack on a cruise-ship [3]. According to Henderson “terror and violent activities are usually political in nature” and port installations appear to have serious vulnerabilities and threats [3]. Various scenarios involving terrorism and cruise ships described in [3] assume that to launch an attack in a port, cyber-terrorists required to be highly skilled and capable of handling specialized maritime operations, such as navigation services and remote control systems.

The work presented in [4], is focused on realizing marine traffic scenarios on vessels' interaction using real-time anomaly detection techniques and practical situation analysis. It produces similar results for any coastal waters and, thus, the research needs to be expanded. In [10], Risk Mitigation from Unmanned Terrorist systems is achieved to reduce the security risk by analyzing the Supply Chain Service and initiating ways to detect threats.

Vulnerabilities identified on the ports' Supply Chain assets have been identified and a quantitative method that sets measures to assess them has been proposed [11]. A vulnerability analysis on maritime operations has been explored in [12].

Attack graph generation and analysis is typically used in prevention techniques, namely, in methods to discover all the possible paths that adversaries can exploit to gain unauthorized access to a system [13]. A research study that derives and analyzes

the impact of empirical attacks targeting Cyber-Physical Systems (CPSs) in a maritime transportation environment has been performed in [14]. A probabilistic approach to explore attack graphs is carried out in [15]. The probability of successful exploits is calculated according to information from libraries of the Common Vulnerability Scoring System and the static security risk is assessed. A distributed attack graph generation algorithm based on a multi-agent system following a depth first search practice shows that performance is improved when agents are used after a certain graph size [16]. A probabilistic model calculating risk security and risk probability regarding dynamic network features is proposed in [17]. A slightly different approach is proposed in [18] adopting a dynamic generation algorithm that returns the top K-paths. A considerable attempt in describing security risk policies and vulnerabilities with the help of semantic technology and conceptual modeling has been reported [19].

Standards and organizational efforts in consensus with these research topics can be also identified: a series of international standards from ISO27 k, [20], to ISO28 k [21] families and practices of IMO, the ISPS code and NIST SP800-30 Rev.1 guidance to detect cyber-assets' vulnerabilities and set metrics for cyber-threat characterization. The literature shows that even though a large amount of cyber perspectives are dedicated to enhance the Security Awareness, there is still a gap in approaching effective solutions. Such solutions must have the ability to characterize the attacker's capabilities, intentions and profile.

Current work presents a dynamic knowledge management practice that uses vulnerability chains to formulate attack path analysis on port Critical Infrastructures, which is structured according to the knowledge-based methodology and security practices described in [6–8] respectively.

3 The Knowledge-Based Approach for Identifying Security Issues on Ports

To identify and analyze security issues of vulnerabilities and cyber-threats on port cyber-assets and discover the possible attack paths, we follow the security approach presented in [7, 8] and adopt knowledge management practices described in [6]. To explore cyber-security issues and discover attack paths on port assets, a specific use-case is detailed. The use-case includes the Vessel Traffic Management System, which is a port Critical Infrastructure supporting the Cruise Service that is a ports' mission critical service. The discovered attack paths are presented into a realistic cyber-terrorist attack scenario to demonstrate how attack paths can be used by intruders to perform a series of cyber-attacks in order to penetrate into the ports' systems and cause damage.

In line with this analysis, a knowledge base has been developed using semantic web standards and formats; specifically the OWL 2 ontology language and the SPARQL query language (Fig. 1). OWL is the most popular and applicable semantic Web language generating tacit knowledge, shown in Fig. 1, from semantic assertions. SPARQL is an excellent semantic query language for unifying and retrieving data in relational databases with other data sources. The ontology engineering follows the ISO 28000:2007 [21] and ISO/IEC 27001:2013 [20] standards supply chain modules along with the ontology structure presented in [6]. Furthermore, security issues on port

Critical Infrastructures will be identified by exploring the asset-centric views of the research methodology presented in [6]. To detect security weaknesses and attack paths on ports infrastructures the following subsequent steps are performed.

3.1 Analysis of the Cruise Vessel Traffic Supply Chain Service (Step 1)

Cruise Vessel Traffic Service aims to inform the Port Authority about the exact arrival time of Cruise Vessels in order to coordinate cruise vessel traffic movements. The scope of this procedure is: (i) to prevent from developing incidents or accidents among cruise vessels while they are reaching a busy destination's port wharf, (ii) to avoid or eliminate pollution and to coordinate response activities. The Vessel Traffic Management Information System (VTMIS) is a Port's Authority automatic marine-traffic monitoring system that generally consists of: a (i) GIS Web Server, connected with a relational database to store and retrieve cruise vessel geolocations and other cruise vessel traffic information, (ii) navigation and remote control services functioning between Port Authority and Cruise Vessels, which are satisfied via satellite communication modules and a TETRA closed radio network, (iii) a Wireless Local Area Network (WLAN) for web communication, (iv) the Vessel Traffic Management Operating System (VTMOS), (v) a Corporate Domain Controller (CDC) responding to security authentication requests within the VTMIS network, (vi) Vessel Traffic Management workstations (VTM workstations) holding a number of web and desktop applications with simple user rights and administrator accounts (admin and super admin), which have proper permission to access and administer all the corporate elements using an SSH Server and an SSH Client, (vii) a router, (viii) a UPS equipment and (ix) power supply that boosts the VTMIS network. External and internal firewalls are installed in all devices of the network to provide security protection.

The analysis of the Vessel Traffic Service is carried out to identify the port's cyber-assets vulnerabilities and security threats. To this end, confirmed or potential/unknown vulnerabilities are identified on the VTMIS cyber-assets deriving information from NIST [22] and CVE's [23] on-line repositories, CVSS open industry standard and automated scanning tools like OpenVas, as presented indicatively in Table 1. Moreover, we assume that a vulnerability identified on an asset has the following main features: (i) Weaknesses in cyber-assets could facilitate the cyber-attacker's mission to compromise their Confidentiality, Integrity or Availability (CIA). Therefore, the impact the asset's identified vulnerabilities may have on the entire system's resilience is considered and it is classified into three levels; complete, partial, none. (ii) The complexity to access the specific cyber-asset cannot also be ignored and is measured on low/medium/high levels, (iii) Cyber-assets authentication policy; it may require either multi-factor authentication confirmation or a single log in or no effort at all to access the system, (iv) the location that a vulnerable system may be accessed. (v) the Common Weakness Enumeration (CWE), to map security vulnerabilities with cyber-threats and have a clearer picture of the security flaw. Cyber asset interdependencies are specified based on two parameters; the type of the dependency and the dependency access vector indicating the network location where the cyber-dependency is achieved, as shown in Table 1.

Table 1. Technical Characteristics, security issues and attributes of indicative cyber-assets functioning for the provision of the Cruise Vessel Traffic Service.

Asset Name	Asset type/Product	Identified Vulnerabilities	CWE/ Identified Threats	Access Vector	Confidentiality	Integrity	Availability	Access Complexity	Authentication
VTM Workstation (A ₁)	Operating System/ Microsoft Windows	CVE-2014-1812 (V _{1A1})	255/Gain Privileges Obtain Infos	N	C	N	N	L	s
		CVE-2016-0117 (V _{3A1})	20/Improper Input Validation	N	C	C	C	M	n
		CVE-2013-7232 (V _{3A2})	89/SQL Injection	N	P	P	P	L	n
GIS Web Server (A ₃)	Web Server/ Apache tomcat	CVE-2012-5568 (V _{1A3})	16/Configuration	N	N	N	P	L	n
		CVE-2013-4444 (V _{2A3})	94/Code Injection	N	P	P	P	M	n
VTMOS (A ₄)	Operating System/ Windows Server 2012 R2 stand.ed.	CVE-2015-0087 (V _{1A4})	200/Information Exposure	N	P	N	N	L	n
		CVE-2015-2554 (V _{2A4})	264/Permissions, Privileges, and Access Controls	L	C	C	C	L	n
Satellite Broadband Device (A ₅)	Hardware/Inmarsat	CVE-2013-6034 (V _{1A5})	255/Credentials Management	N	C	C	C	L	n

3.2 Cyber Threat Identification (Step 2)

Cyber-threats are recognized either from online repositories, using crowdsourcing or from social media. Table 1 presents indicative examples on VTMS assets. The ontology adopts the triple-entity structure to map cyber-assets with the addressed vulnerabilities and threats. The Common Weakness Enumeration is the node for matching vulnerability with cyber-threats. Such information is derived from the ontology using SPARQL queries, an example of which is presented in the following step.

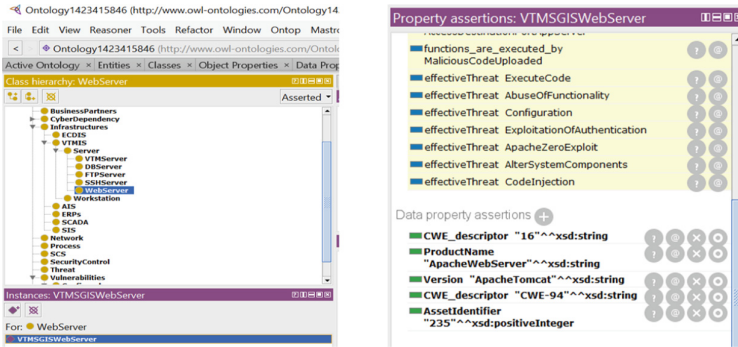


Fig. 1. A screenshot of the OWL ontology structure and properties, depicting threats of the GIS web server as tacit knowledge from the object property assertion “effective threat”.

3.3 Attack Path Detection (Step 3)

Taking into account the attack prediction theory described in [7, 8] we discover possible attack paths that terrorists could take advantage of to commit a cyber-attack that could threaten the ports’ safety and integrity. In this step, vulnerabilities addressed on the port’s assets are observed. To discover potential attack paths, characteristics described in [22, 23] must be considered: (i) Explore the Attacker’s Profile regarding the relationship with the organization: for example to estimate if he is an outsider or an insider, recognize his motivation (terrorist/hacktivist/insider/disgruntled employee) and his skills. (ii) Discover the assets that can be “Entry points” for an attacker. This depends on the specific attacker’s characteristics. For example, assuming that the attacker is an *an outsider*, that means that the attacker can attack only “remotely”. According to Table 1, every asset can be an entry point by exploiting the vulnerability that has the value “Network” in the Access Vector attributes. Consequently, every vulnerability can be exploited *except* the vulnerability “CVE-2015-2554” found on the asset VTMOs, which can be exploited only locally. (iii) Consider whether the attackers can exploit vulnerabilities identified on different assets (vulnerability chains) that are interdependent, giving them consequently the opportunity to perform a series of cyber-attacks and thus estimate how deeply the potential attacker could penetrate into the organization’s network. Eventually, we can use the developed Knowledge Base, to extract implicit information with SPARQL queries or OWL declarations. For example, to request, vulnerabilities of category “remote code execution” detected in the VTM Workstation we explore the following SPARQL query in which the standard prefix declarations (rdf, owl, rdfs and xsd) are omitted:

```

PREFIX f: <http://www.owl-ontologies.com/Ontology1423415846.owl#>
SELECT ?x ?z ?y
WHERE {?x ?z "Remote Code
Exection"^^<http://www.w3.org/2001/XMLSchema#string> . ?x f:isExploitableOn
f:VTMWorkstation }
(1)

```

Query results: x = CVE-2016-0117 z = VulnerabilityCategory

Attack Path Analysis Example. Attack paths have been generated according to the attack prediction theory dedicated work [8] and are visualized in Fig. 2. Taking into account the vulnerability attributes and threats shown in Table 1, vulnerabilities on different interconnected assets have been recognized that could be potentially exploited by cyber-attackers to initiate a series of cyber-attacks and compromise a considerable part of the port's Critical Infrastructure. Table 2 presents the vulnerability chains that could be exploited to take advantage of the Cruise Service assets. The cyber-attack scenario ensuing is fabricated according to the attack-path results extracted from the proposed knowledge base. It is, though, a realistic use-case that can be implemented by cyber-terrorists to compromise the current VTMS of the Cruise Service. The cyber-attack scenario is used as an indicative example to illustrate the attack paths recognized from the utilization of vulnerabilities and the effective implementation of security threats detected in the targeted assets.

Description of the Terrorist Attack Scenario. Terrorists aim to launch cyber-attacks to access and cause damage to cyber-assets of the Port Authority infrastructure that hosts the Vessel Traffic System of the Cruise Service. To achieve this, they carry out a series of cyber-attacks, as presented in the following. A skilled hacker of the terrorist group sends phishing e-mails to Port Cruise Terminal corporate staff, asking them to click on a link that directs them to a fraudulent web page, which seems legitimate. A VTMS user of the port's Cruise Service is convinced that the e-mail comes from an official partner, thus, he opens it and clicks it and gets automatically redirected to a crafted .pdf document, which enables the adversary, by exploiting the CVE-2016-0117 vulnerability, to gain the control of the VTM workstation with the port employee's simple user rights. Then, he authenticates as simple domain user, exploits the CVE-2014-1812 vulnerability found on the VTM workstation and obtains sensitive information with elevated privileges of the VTMS administrator's account. The adversary can now leverage his access through a web browser that allows him to reach and exploit the CVE-2013-7232 vulnerability, found on the GIS Vessel Traffic software which delivers cruise vessels' geolocation services. Hence, the attacker can: (i) initiate arbitrary SQL statements including values dependent on the adaptor's live-ship map default type and the corresponding database, resulted in altering the cruise vessels' geolocation and (ii) read the credentials of the GIS Web Server in which the GIS Vessel Traffic software is deployed. The GIS Web Server, installed on the VTMOs, allows the adversary to log in as an authenticated user and exploit the CVE-2013-4444 vulnerability found on it succeeding to obtain the control of the VTMOs. Thereafter, the terrorist hacker can utilize the CVE-2015-0087 vulnerability to reveal secret data from kernel memory of the VTMOs and get access to it. The Satellite Broadband terminal, used for voice communication with the cruise vessels, and the VTMOs are located in the same WLAN. The adversary exploits the CVE-2013-6034 flaw of the Satellite terminal and obtains login access using its hardcoded credentials and eventually compromise the satellite ground device, causing to shut it down. This could lead to multiple glitches and disruptions bringing confusion and chaos both in the port facilities and water adjacent area. For example, a ship collision may cause human casualties, serious damages to the vessels' hull and environmental harm.



Fig. 2. An attack-path query depicting how skilled hacker terrorists can launch a series of cyber-attacks to the VTMS of the Cruise Service infrastructure; starting from a simple domain user workstation, ending up to satellite modem penetration and closure.

Table 2. Attack Path results for the VTMS cyber-assets of the Cruise Service.

Asset Chain (A1 → A2 → A3 → A _x)	Vulnerability Chain (V1 → V2 → V _x)	Assets' Chain Name	Assets' Chain Vulnerabilities
VTM workstation → GIS Vessel Traffic software → GIS Vessel Traffic Web Server → VTMO → Satellite Broadband Device	V _{3A1} :CVE-2016-0117 → V _{1A1} :CVE-2014-1812 →	VTM	Remote Code Execution (V _{3A1})
		Workstation	Gain privileges, Obtain Info (V _{1A1})
	V _{3A} :CVE-2013-7232 →	Vessel Traffic software	Execute Code, Sql Injection (V _{3A2})
	V _{2A3} :CVE-2013-4444 →	GIS Web Server	Remote Code Execution (V _{2A3})
	V _{1A4} :CVE-2015-0087 →	VTMO	Bypass a restriction or similar-obtain information (V _{1A4})
	V _{1A5} : CVE-2013-6034	Satellite Broad-band Device	unauthorized disclosure of information (V _{1A5})

4 Conclusions

This work adopted a knowledge-based methodology using inference rules and mechanisms to develop ontology models and extract new knowledge regarding port' Critical Infrastructures. Precisely, the outcome of this work is the asset analysis, vulnerabilities and threats identification, recognition of interrelations among assets–vulnerabilities–threats and the discovery of vulnerability chains and attack paths. Semantic web technologies are used, such as SPARQL queries and OWL constraints, because ontologies and semantic web practices can dynamically retrieve, store and easily update information wherever required. To better illustrate these results, we took into account the case-study of a mission critical service, namely the Cruise Vessel Traffic Service. Once vulnerability chains and attack paths are discovered, we formulated a terrorist attack scenario, based on the information acquired, to highlight the impact of the exploitation of these flaws. Future work could focus on the further expansion of the ports' assets security knowledge, by combining cyber-physical models and cyber security issues.

Acknowledgements. This work has been partially supported by the University of Piraeus Research Centre and the European Union's Horizon 2020 project "SAURON" under grant agreement No 740477 addressing the topic CIP-01-2016-2017. The authors would like to thank all project members for their valuable insights. Finally, special thanks to the University of Piraeus, Research Centre for its continuous support.



References

1. The Risk of Cyber-Attack to the Maritime Sector. http://www.ahcusa.org/uploads/2/1/9/8/21985670/the_risk_of_cyber-attack_to_the_maritime_sector-07-2014.pdf. Accessed 11 Apr 2018
2. Roell, P.: Maritime Terrorism. A threat to world trade? <https://www.files.ethz.ch/isn/110282/MaritimeTerrorism.pdf>. Accessed 11 Apr 2018
3. Bowen, C., Fidgeon, P., Page, S.J.: Maritime tourism and terrorism: customer perceptions of the potential terrorist threat to cruise shipping. *CI in Tourism* **17**(7), 610–639 (2014)
4. Shahir, H.Y., Glasser, U., Shahir, A.Y., Wehn, H.: Maritime situation analysis framework: vessel interaction classification and anomaly detection. In: 2015 IEEE International Conference on Big Data, Santa Clara, CA, pp. 1279–1289 (2015). <https://doi.org/10.1109/BigData.2015.7363883>
5. Bueger, C.: What is maritime security? *Mar. Policy* **53**, 159–164 (2015)
6. Kalogeraki, E.-M., Apostolou, D., Polemi, N., Papastergiou, S.: Knowledge management methodology for identifying threats in maritime/logistics supply chains. In: Durtst, S., Evangelista, P. (eds.) (SI) "Logistics Knowledge Management: State of the Art and Future Perspectives", Knowledge Management Research and Practice Journal. Taylor and Francis (2018). ISSN: 1477-8238 (Print). ISSN: 1477-8246. <https://doi.org/10.1080/14778238.2018.1486789>
7. Papastergiou, S., Polemi, N.: MITIGATE: a dynamic supply chain cyber risk assessment methodology. In: Yang, X.-S., Nagar, A.K., Joshi, A. (eds.) *Smart Trends in Systems, Security and Sustainability*. LNNS, vol. 18, pp. 1–9. Springer, Singapore (2018). https://doi.org/10.1007/978-981-10-6916-1_1

8. Polatidis, N., Pimenidis, E., Pavlidis, M., Papastergiou, S., Mouratidis, H.: From product recommendation to cyber-attack prediction: generating attack graphs and predicting future attacks. *Evol. Syst.*, 1–12 (2018). Springer-Verlag GmbH, Germany. ISSN: 1868-6478. <https://doi.org/10.1007/s12530-018-9234-z>
9. SAURON Homepage. <https://www.sauronproject.eu/>. Accessed 11 Apr 2018
10. Patterson, M.R., Patterson, S.J.: Unmanned systems: an emerging threat to waterside security: bad robots are coming. In: 2010 International WaterSide Security Conference, Carrara, pp. 1–7 (2010). <https://doi.org/10.1109/WSSC.2010.5730271>
11. Wagner, S.M., Neshat, N.: Assessing the vulnerability of supply chains using graph theory. *Int. J. Prod. Econ.* **126**(1), 121–129 (2010)
12. Liu, H., Tian, Z., Huang, A., Yang, Z.: Analysis of vulnerabilities in maritime supply chains. *Reliab. Eng. Syst. Saf.* **169**, 475–484 (2018)
13. Ou, X., Singhal, A.: Attack graph techniques. In: Ou, X., Singhal, A. (eds.) *Quantitative Security Risk Assessment of Enterprise Networks*. SpringerBriefs in Computer Science, pp. 5–8. Springer, New York (2012). https://doi.org/10.1007/978-1-4614-1860-3_2
14. Bou-Harb, E., Kaisar, E.I., Austin, M.: On the impact of empirical attack models targeting marine transportation. In: *Proceedings of the 5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems, MT-ITS 2017, Naples*, pp. 200–205 (2017). <https://doi.org/10.1109/MTITS.2017.8005665>
15. Gao, N., He, Y., Ling, B.: Exploring attack graphs for security risk assessment: a probabilistic approach. *Wuhan Univ. J. Nat. Sci.* **23**(2), 171–177 (2018)
16. Kaynar, K., Sivrikaya, F.: Distributed attack graph generation. *IEEE Trans. Dependable Secure Comput.* **13**(5), 519–532 (2016)
17. Almohri, H.M.J., Watson, L.T., Yao, D., Ou, X.: Security optimization of dynamic networks with probabilistic graph modeling and linear programming. *IEEE Trans. Dependable Secure Comput.* **13**(4), 474–487 (2016)
18. Bi, K., Han, D., Wang, J.: K maximum probability attack paths dynamic generation algorithm. *Comput. Sci. Inf. Syst.* **13**(2), 677–689 (2016)
19. Ghiran, A.-M., Buchmann, R.A., Osman, C.-C.: Security requirements elicitation from engineering governance, risk management and compliance. In: Kamsties, E., Horkoff, J., Dalpiaz, F. (eds.) *REFSQ 2018*. LNCS, vol. 10753, pp. 283–289. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-77243-1_17
20. ISO/IEC 27001:2013: Information technology – Security techniques – Information security management systems – Requirements, ISO/IEC
21. ISO 28000:2007: Specification for security management systems for the supply chain, Geneva, Switzerland: ISO/IEC
22. National Institute of Standards and Technology. <https://nvd.nist.gov/>. Accessed 11 Apr 2018
23. Common Vulnerabilities and Exposures. <https://cve.mitre.org/>. Accessed 11 Apr 2018



Prognosis of Thyroid Disease Using MS-Apriori Improved Decision Tree

Yuwei Hao^{1,2} , Wanli Zuo^{1,2}, Zhenkun Shi^{1,2} , Lin Yue³,
Shuai Xue⁴, and Fengling He^{1,2}

¹ College of Computer Science and Technology, Jilin University,
Changchun 130012, China

shizk14@mails.jlu.edu.cn

² Key Laboratory of Symbol Computation and Knowledge Engineering,
Jilin University, Ministry of Education, Changchun 130012, China

³ School of Computer Science and Information Technology,
Northeast Normal University, Changchun 130117, China

⁴ The First Hospital of Jilin University, Changchun 130021, China

Abstract. The lymph nodes metastasis in the papillary thyroid microcarcinoma (PTMC) can lead to a recurrence of cancer. We hope to take preventive measures to reduce the recurrence rate of the thyroid cancer. This paper presents a decision tree improved by MS-Apriori for the prognosis of lymph node metastasis (LNM) in patients with PTMC, called MsaDtd (Decision tree Diagnosis based on MS-Apriori). The method converts the original feature space into a more abundant feature space, MS-Apriori is used to generate association rules that consider rare items by multiple supports and fuzzy logic is introduced to map attribute values to different subintervals. Then, we filter the ranked rules which consider positive and negative tuples. We improve accuracy through deleting disturbance rules. At last, we use the decision tree to predict LNM by analyzing the affiliation between the instance and rules. Clinical-pathological data were obtained from the First Hospital of Jilin University. The results show that the proposed MsaDtd achieves better prediction performance than other methods on the prognosis of LNM.

Keywords: MS-Apriori · Decision tree · Medical mining · Disease predication

1 Introduction

Artificial intelligence (AI) has recently gained a tremendous advance in various applications, e.g., autonomous driving, big data, pattern recognition, intelligent search, image understanding, automatic programming, and robotics [1]. These applications also inspire AI technique to develop and innovate, in a way. The increasing availability of healthcare data and rapid development of big data analytic methods have made possible the recent successful applications of AI in healthcare [2]. Machine Learning as one of the core technologies of AI has been widely used in all walks of life. In recent years, the healthcare industry produces a huge amount of digital data by utilizing information from all sources of healthcare data such as Electronic Health Records [3] and Personal Health

Records [4]. At the same time, machine learning is well poised to assist clinical researchers in deciphering complex predictive patterns in healthcare data [5]. All of these provides the basis of the prognostication of diseases with Machine Learning technique.

Indeed, the incidence of thyroid cancer has nearly tripled since 1975 [6]. In PTMC, the prevalence of subclinical CLNM has been detected as 30%–65% [7]. And PTMC can lead to a recurrence of cancer. Therefore, it is urgent to introduce machine learning into the field of Thyroid Disease. To solve the prognostic problem of Thyroid Disease, we propose a disease diagnosis model, and apply it to thyroid disease diagnosis in the First Hospital of Jilin University.

The technical contributions done in this paper are summarized as follows:

1. We propose an algorithm MsaDtd that converts the original characteristic space into a larger characteristic space and improved decision tree algorithm for disease diagnosis to predict LNM in patients with PTMC.
2. We use MS-Apriori to obtain composite features, taking into account rare items by setting multiple minimum supports (MIS), and introduce fuzzy logic to deal with continuous attributes, aiming to avoid the cost of producing large frequent items.
3. We use 5425 Clinical-pathological data of PTMC patients in the First Hospital of Jilin University to validate MsaDtd. Experimental analysis indicates that the algorithm predicts LNM effectively and accurately.

2 Related Work

Prediction of thyroid diseases using machine learning has been an ongoing effort in recent years. Chen et al. [8] presented a three-stage expert system based on a hybrid support vector machines (SVM). It combined feature selection and parameter optimization, the developed FS-PSO-SVM expert system achieved excellent performance in distinguishing among hyperthyroidism, hypothyroidism and normal ones. Makas et al. [9] developed seven distinct sorts of Neural Networks to identify the thyroid disease. And used particle swarm optimization (PSO), artificial bee colony (ABC) and migrating birds optimization (MBO) algorithms retrained the network. The accuracy of the network developed outperformed the similar studies. Pourahmad et al. [10] used a back propagation feedforward neural networks to diagnose the malignancy in thyroid tumor. Thirteen batch learning algorithms were investigated and three different numbers of neuron in hidden layer were compared to achieve the best performance. Kaya et al. [11] applied Extreme Learning Machine (ELM) to the diagnosis of thyroid disease. This study indicated the classification and speed of ELM were higher than other machine learning methods. Maysanjaya et al. [12] used Multilayer Perceptron method to identify the type of thyroid (normal, hypothyroid, hyperthyroid) with WEKA tool. The accuracy of the prediction was as high as 96.74%.

Researchers have done much research on solving the problem of thyroid diseases diagnosis. But there are few studies on the prognosis of LNM in patients with PTMC. The prognosis of LNM is essential to prevent recurrence of cancer. For the above situation, this paper designs an intelligent decision model MsaDtd to predicts lymph node metastasis (LNM) in patients with PTMC.

3 The Prognosis Algorithm Based on MS-Apriori and Decision Tree

We design a disease diagnostic algorithm by mapping the prognosis of LNM in patients with PTMC to a binary classification problem. The symptoms of patients are mapped to independent variables $\mathbf{u} = (u_1; u_2; \dots; u_d)$ and diagnostic results are mapped to dependent variables $y \in \{0, 1\}$.

3.1 MS-Apriori Rule Mining

Apriori play a major role in identifying frequent itemset and deriving rule set out of it [13]. Using Apriori results in a shortage when mining rare knowledge patterns of rare events, due to the entire database only set one minimum support. To solve this problem, we use MS-Apriori setting MIS for different items.

For attribute value, this paper introduces fuzzy logic to map attribute values to different subintervals through membership function, aiming to avoid the cost of producing large frequent items.

The association rule mining process is as follow. An item type v_i is defined as each value type under each attribute in clinical-pathological data. The set of items in the whole database is I shown in Eq. (1) and the item type set is V shown in Eq. (2).

$$I = \{a_1, a_2, \dots, a_m\} = IA_1 \cup IA_2 \cup \dots \cup IA_d, m = n * d \tag{1}$$

$$V = \{v_i\}, i = 1, 2, \dots, u \tag{2}$$

There are m items, u item types and d -dimension features in the whole database. $IA = \{a_i\}$ ($i = 1, 2, \dots, n$) represents the fuzzy itemset under an attribute. We specify that different attribute values under the same attribute do not belong to the same frequent itemset. The items in a frequent itemset should meet the condition shown in Eq. (3).

$$a_x \cap a_y = \emptyset, a_x \in IA_i, a_y \in IA_j, i = j \tag{3}$$

In addition, the support of frequent itemset is the smallest MIS of items in the frequent itemset. The frequent itemset is defined as Eq. (4). The MIS of frequent itemset c is defined as Eq. (5). The MIS of the item is defined as Eq. (6).

$$c = \{a_1, a_2, \dots, a_k\}, 1 \leq k \leq d \tag{4}$$

$$MIS(c) = \min(MIS(a_1), MIS(a_2) \dots MIS(a_k)) \tag{5}$$

$$MIS(v_i) = \frac{v_i \cup LM_{yes}}{N} \tag{6}$$

v_i represents an item, corresponding a value type in clinical-pathological data. LM_{yes} represents the label of patients is lymph node metastasis. N is the total number of instances. The probability of item v_i and item LM_{yes} appear in the same frequent itemset is set to the MIS of v_i .

The frequent item c_j is converted to rule $Rule_j$ shown in Eqs. (7) and (8).

$$c_j : a_1 \cup a_2 \cup \dots \cup LM_{yes}/LM_{no} \quad (7)$$

$$Rule_j \rightarrow LM_{yes}, Rule_j : a_1 \cup a_2 \cup \dots \cup a_{k-1} \quad (8)$$

We rank the rule by cosine measure and delete disturbance rules by defining a threshold. The cosine measure of positive tuple rules is defined by Eq. (9).

$$\text{cosine}(Rule_j, LM_{yes}) = \frac{P(Rule_j \cup LM_{yes})}{\sqrt{P(Rule_j) * P(LM_{yes})}} \quad (9)$$

$P(Rule_j \cup LM_{yes})$ represents the probability that $Rule_j$ and LM_{yes} belong to the same frequent item. The cosine measure of negative tuple rules is defined as Eq. (10).

$$\text{cosine}(Rule_j, LM_{no}) = \frac{P(Rule_j \cup LM_{no})}{\sqrt{P(Rule_j) * P(LM_{no})}} \quad (10)$$

Algorithm 1 outlines the process of rule mining by MS-Apriori. SDC is used to limit a rare item and a common item appear in the same frequent item. $threshold$ is used to delete disturbance rules.

Algorithm 1:MS-Apriori rule mining algorithm

Input: LNM Dataset $D = \{(u_i, y_i)\}, i = (1, 2, \dots, n), y_i \in \{0, 1\}$ which contains training instances $u = (u_1; u_2; \dots; u_d)$ and their associated diagnostic labels $y \in \{0, 1\}$, membership functions $\{\delta_m(x)\}, m = (1, 2, \dots, d)$, $SDC, threshold$

Output: sorted rule set which removed disturbance rules:

$R = \{rule \mid \text{cosine}(rule) \geq threshold\}$

1. Convert D to T by converting each attribute value in D to a_i by $\delta_m(x)$
 2. Compute multiple minimum supports for each item by $MIS(v_i) = \frac{v_i \cup LM_{yes}}{N}$
 3. Generate frequent 1 itemset $F_1 = \{c \mid c \in T, c.count \geq MIS(c)\}$
 4. **for** ($k = 2; F_{k-1} \neq \emptyset; k++$)
 5. $C_k = \text{gen_candidate}(F_{k-1}, SDC)$
 6. **for each** transaction t in T **do**
 7. Storage t in C_k when t is the subset of C_k
 8. **for each** candidate c in C_k **do**
 9. $c.count++$
 10. Generate $F_k = \{c \in C_k \mid c.count \geq MIS(c)\}$
 11. Generate $\{F_k\}$ ($k=1, 2, \dots, k$)
 12. **for each** c in $\{F_k\}$ **do**
 13. Convert c to Diagnosis rules $rule$
 14. ranking $rule$ and delete the rule when $\text{cosine}(rule) < threshold$
-

3.2 Decision Tree Construction

We obtain the sorted rule set $R = \{rule | cosine(rule) \geq threshold\}$ which is closely related to LMN diagnosis, through mining association rules in clinical-pathological data. Next, we build a decision tree which is used to predict LNM.

Through converting each rule in rule set R to the candidate attributes of the decision tree, the algorithm generates attribute set A . To determine which rule is selected as the splitting attribute in the process of classification, information gain is used as a decision criterion. When an instance contains all items needed in $rule_i$, this rule can be applied to this instance. $rule_i$ as a new attribute, its attribute value is LM_{yes}/LM_{no} . If the rule is positive tuple rule, the value of $rule_i$ is LM_{yes} after applying the rule. If the rule is negative tuple rule, the value of $rule_i$ is LM_{no} after applying the rule. Otherwise, the rule cannot be applied, the value is No. The dataset D is converted to $S = \{(x_i, y_i)\}, i = (1, 2, \dots, n), y_i \in \{0, 1\}$. The labels of the dataset are LNM and normal. We mark it as S_1 and S_0 . The information entropy of S is defined as Eq. (11).

$$H(S) = - \sum_{i=1}^2 p_i \log_2 p_i \quad (11)$$

$$p_i = \frac{S_i}{S}, i = 1, 2 \quad (12)$$

Where p_i represents the probability that $x_i \in S$ belongs to a class S_i , and is estimated by Eq. (12). The information gain for attribute $r \in A$ at node N is defined as Eq. (13).

$$Gain(S, r, N) = H(S) - \sum_{j=0}^1 \frac{S_j}{S} H(S_j) \quad (13)$$

The attribute with the maximum information gain is selected as the splitting attribute at node N . The instances are recursively partitioned into smaller subsets through analyzing the affiliation between instances and the rules mined by MS-Apriori. When all the subsets belong to a single class, or there are no instance or attribute can be used to partition, a model used to predict LNM is constructed.

4 Experiments

4.1 Data Pre-processing

This study is conducted in the Thyroid Surgery of the First Hospital of Jilin University. A total of 5425 patients with PTMC who underwent thyroidectomy with neck dissection from 2011 to 2015 are studied. Among the 5254 patients, there are 4855 cases met the criteria, including 323 cases treated lateral neck dissection.

Features used in this study include gender, age, capsule invasion (CI), maximum tumor diameter (MTD), multifocal, Hashimoto thyroiditis (HT), Central lymph node

Table 1. Description of feature

Features	Gender	Age	CI	MTD	Multifocal	HT	CN	LN
range	Male/female	12–82	0–1	0/1	0/1	0/1	0–34	0–87

number (CN). These features are shown in Table 1. For LLNM, adding two additional features, CLNM and lateral lymph node number (LN).

In this paper, we use the box plot to analyze data. We identify noise data by IRQ and set the value of it as null. Because box plot identifies abnormal values more objective and quartiles have a certain degree of robustness. For the missing values, in order to avoiding the loss of information by deleting. We should speculate missing values based on the majority of the existing data. We use mean/mode imputation (MMI) to deal with missing values. A bias occurs when we use it to train a predictive model, because of the unbalanced data. To solve the problem of skewed data, we use balancing techniques. The techniques we use is on CNLM dataset is KNN-NearMiss-2, a kind of supervised under-sampling techniques based on K-nearest neighbor. For LLNM dataset, SMOTE over-sampling technique is used, due to the small number of instances.

4.2 Results and Discussion

The proposed predictor is applied to the Clinical-pathological data of the First Hospital of Jilin University. To illustrate the performance of MsaDtd, we compare MsaDtd with a range of baseline algorithms, including Decision Tree (DT), Support Vector Machines (SVM), Logistic regression (LR), Bernoulli Bayes (BNB). We use 10-fold cross-validation to valid MsaDtd algorithm on CLNM dataset and LLNM dataset.

Table 2. Performance comparison with baseline algorithms on CLNM dataset

	Accuracy	Precision	Recall	F ₁	AUC
MsaDtd	76.09%	72.16%	63.63%	72.63%	82.06%
DT	73.62%	67.57%	72.44%	73.94%	74.13%
SVM	71.03%	65.86%	63.22%	68.54%	75.34%
LR	70.58%	64.97%	67.27%	69.56%	75.37%
BNB	59.05%	55.38%	62.88%	60.54%	62.32%

Table 3. Performance comparison with baseline algorithms on LLNM dataset

	Accuracy	Precision	Recall	F ₁	AUC
MsaDtd	87.21%	82.75%	85.86%	86.85%	88.37%
DT	83.70%	78.54%	83.95%	83.76%	83.20%
SVM	79.19%	71.72%	91.02%	81.40%	86.08%
LR	78.79%	72.11%	84.43%	79.80%	87.31%
BNB	75.08%	68.53%	82.38%	76.74%	82.42%

Tables 2 and 3 shows the results of various algorithms on CLNM dataset and LLNM dataset, respectively. As we can see, on CLNM dataset, MsaDtd algorithm

achieves the results with Accuracy, Precision, Recall, F_1 and AUC values are 76.09%, 72.16%, 63.63%, 72.63%, and 82.06%. High prediction accuracy of 76.09% is obtained for MsaDtd algorithm. The accuracy of the improved decision tree is higher than the traditional decision tree and other classifiers. The accuracy of improved decision tree MsaDtd increased by 2.47% compared with the traditional decision tree. On LLNM dataset, the average prediction Accuracy, Recall, Precision, F_1 , and AUC of MsaDtd are 87.21%, 82.75%, 85.86%, 86.85% and 88.37%. Our method outperforms the traditional decision tree in all aspects. The Accuracy, Recall, Precision, F_1 , and AUC increased by 3.51%, 4.21%, 1.91%, 3.09% and 5.17% comparing to the decision tree. Our method has the highest Accuracy, Recall, Precision and AUC among the methods we compared.

Figures 1 and 2 shows a plot of the ROC curves derived from MsaDtd and various baseline algorithms on different dataset. One CLNM dataset, it is higher 6.69% than LR which having the highest ROC area among baseline algorithms. On LLNM dataset, the Roc area of MsaDtd is 88.37%, which is higher than all of the methods mentioned. The above results show the superior performance of the prediction we proposed.

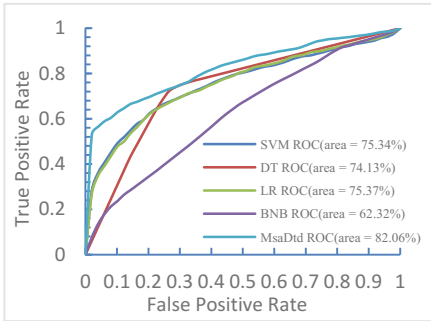


Fig. 1. ROC curve comparison with baseline algorithms on CLNM dataset

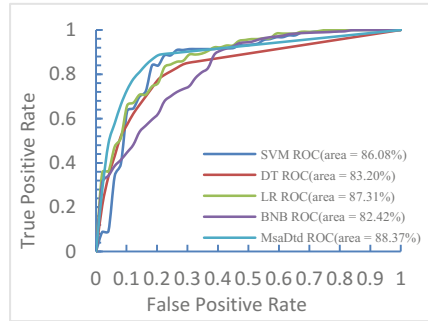


Fig. 2. ROC curve comparison with baseline algorithms on LLNM dataset

Table 4. Performance comparison with DeepPPI on CLNM and LLNM dataset

Dataset	Method	Accuracy	Precision	Recall	F_1	AUC
CLNM	MsaDtd	76.09%	72.16%	63.63%	72.63%	82.19%
	DeepPPI	65.66%	63.78%	73.96%	68.28%	74.71%
LLNM	MsaDtd	87.21%	82.75%	85.86%	86.85%	89.15%
	DeepPPI	81.83%	76.22%	92.10%	83.05%	87.09%

To our Knowledge, there is almost no one proposed the specialized algorithm for the prognosis of lymph node metastasis (LNM) in patients with PTMC in recent years, so we compare our method with a classification algorithm DeepPPI-Con [14] which achieves superior performance in Protein-Protein Interactions. The results shown in Table 4 indicate that our method is significantly superior to DeepPPI. The Accuracy,

Precision, F_1 and AUC of MsaDtd are 10.43%, 8.38%, 4.35% and 7.48% higher than DeepPPI on CLNM dataset. They are increased by 5.38%, 6.53%, 3.8% and 2.06% comparing to DeepPPI on LLNM dataset.

5 Conclusion

In this paper, we propose an algorithm MsaDtd which improved decision tree with MS-Apriori and applied to the prognosis of thyroid disease through establishing a predictor to predict LNM in patients with PTMC. Fuzzy logic is introduced to handles continuous attributes, preventing to generate too many frequent items. Sorting and filtering rules mined by MS-Apriori used to avoid generate distractions, aiming to improve the prediction accuracy. Through the application of rules, the algorithm obtains new features to transform feature space, making full use of composite features. This improves the robustness and generalization capabilities of our algorithm. Building a decision tree and predicting thyroid disease by analyzing the affiliation between instances and rules to make the effective prediction. Clinicians can use the information given by predictor to adopt specific protocols throughout treatment. For the patients prone to LNM, clinicians should take customized interventions to reduce the risk of cancer recurrence.

Acknowledgement. Project supported by the Nature Science Foundation of Jilin Province (No. 20180101330JC), the National Nature Science Foundation of China (No. 60973040), the Fundamental Research Funds for the Central Universities (No. 2412017QD028), China Post-doctoral Science Foundation (No. 2017M621192), the Scientific and Technological Development Program of Jilin Province (No. 20180520022JH).

References

1. Fan, M., Hu, J., Cao, R., et al.: A review on experimental design for pollutants removal in water treatment with the aid of artificial intelligence. *Chemosphere* **200**, 330–343 (2018)
2. Jiang, F., Jiang, Y., Zhi, H., et al.: Artificial intelligence in healthcare: past, present and future. *Stroke Vasc. Neurol.* **2**(4), 230–243 (2017)
3. Jiang, H., Zhang, Z., Tao, L.: A semantic-based EMRs integration framework for diagnosis decision-making. In: Buchmann, R., Kifor, C.V., Yu, J. (eds.) KSEM 2014. LNCS (LNAI), vol. 8793, pp. 380–387. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-12096-6_34
4. Fang, R., Pouyanfar, S., Yang, Y., et al.: Computational health informatics in the big data age: a survey. *ACM Comput. Surv.* **49**(1), 12 (2016)
5. Vemulapalli, V., Qu, J., Garren, J.M., et al.: Non-obvious correlations to disease management unraveled by Bayesian artificial intelligence analyses of CMS data. *Artif. Intell. Med.* **74**, 1–8 (2016)
6. Tomaszewski, J.J., Uzzo, R.G., Egleston, B., et al.: Coupling of prostate and thyroid cancer diagnoses in the United States. *Ann. Surg. Oncol.* **22**(3), 1043–1049 (2015)
7. Akin, Ş., Yazgan, A.D., Akin, S., et al.: Prediction of central lymph node metastasis in patients with thyroid papillary microcarcinoma. *Turk. J. Med. Sci.* **47**(6), 1723 (2017)
8. Chen, H.L., Yang, B., Wang, G., et al.: A three-stage expert system based on support vector machines for thyroid disease diagnosis. *J. Med. Syst.* **36**(3), 1953–1963 (2012)

9. Makas, H., Yumusak, N.: A comprehensive study on thyroid diagnosis by neural networks and swarm intelligence. In: International Conference on Electronics, Computer and Computation, pp. 180–183. IEEE, Ankara (2014)
10. Pourahmad, S., Azad, M., Paydar, S.: Diagnosis of malignancy in thyroid tumors by multi-layer perceptron neural networks with different batch learning algorithms. *Glob. J. Health Sci.* **7**(6), 46–54 (2015)
11. Kaya, Y.A.: Fast intelligent diagnosis system for thyroid diseases based on extreme learning machine. *Arch. Otolaryngol. Head Neck Surg.* **15**(1), 41–49 (2014)
12. Maysanjaya, I.M.D., Nugroho, H.A., Setiawan, N.A.: A comparison of classification methods on diagnosis of thyroid diseases. In: International Seminar on Intelligent Technology and ITS Applications, pp. 89–92. IEEE, Surabaya (2015)
13. Chaudhary, R., Sharma, S., Sharma, V.K.: Improving the performance of MS-Apriori algorithm using dynamic matrix technique and map-reduce framework. *Int. J. Innov. Res. Sci. Technol.* **2**(5), 2349–6010 (2015)
14. Du, X., Sun, S., Hu, C., et al.: DeepPPI: boosting prediction of protein-protein interactions with deep neural networks. *J. Chem. Inf. Model.* **57**(6), 1499–1510 (2017)



Stock Price Prediction Using Time Convolution Long Short-Term Memory Network

Xukuan Zhan, Yuhua Li^(✉), Ruixuan Li, Xiwu Gu, Olivier Habimana,
and Haozhao Wang

School of Computer Science and Technology,
Huazhong University of Science and Technology, Wuhan 430074, China
{zhanxk, idcliyuhua, rxli, guxiwu, hz.wang}@hust.edu.cn, habolivier@ymail.com

Abstract. The time series of stock prices are non-stationary and non-linear, making the prediction of future price trends much challenging. Inspired by Convolutional Neural Network (CNN), we make convolution on the time dimension to capture the long-term fluctuation features of stock series. To learn long-term dependencies of stock prices, we combine the time convolution with Long Short-Term Memory (LSTM), and propose a novel deep learning model named Time Convolution Long Short-Term Memory (TC-LSTM) networks. TC-LSTM can obtain the stock longer data dependence and overall change pattern. The experiments on two real market datasets demonstrate that the proposed model outperforms other three baseline models in the mean square error.

Keywords: Time convolution · Long Short-Term Memory (LSTM)
Stock price prediction

1 Introduction

The prediction of stock prices is a challenging task because of highly volatile and non-stationary nature of market [1]. Even more, predicting the stock prices in short term range relies on short-term trends, while getting the long term range prediction relies on discovering different trading patterns. Therefore, it is important to discover the long-term change pattern of stocks. This inspires us to find the way how to get the overall features of the market. We can learn the market fluctuation features by convolving time series data in the time dimension, and we call it time convolution.

On the other hand, given the non-linearity and non-stationarity of stock prices, Recurrent Neural Network (RNN) has emerged as a candidate to learn temporal patterns such as price fluctuation. However, due to the vanishing gradient problem, RNN could fail to learn long-term dependencies in a time series [8]. Long Short Term Memory (LSTM) [10], a variant of RNN, was proposed to address this problem. The network structure of LSTM is composed of several

types of memory gates, which enables LSTM to capture the long-term dependency of stock prices at different moments. Inspired by the gating architecture and time convolution, we present the Time Convolution Long Short-Term Memory (TC-LSTM) recurrent network to predict the stock price.

The main contributions of this paper are as follows:

1. We create a new deep learning model TC-LSTM that combines time convolution with LSTM to predict the stock prices. The new model has the ability of time convolution to extract regional features of stock price series and the advantage of LSTM to process sequential data.
2. We conduct experiments on the real market data to test the proposed TC-LSTM network. The results demonstrate that TC-LSTM exceeds other baseline models in the predictive accuracy.

The remainder of this paper is organized as follows. We first review the related work in Sect. 2, followed by the presentation of the model architecture in Sect. 3. Experiment results on the stock price data sets are demonstrated in Sect. 4. The conclusions are given in Sect. 5.

2 Related Work

There are many approaches to do stock price prediction [4, 9, 12]. [3] presents extensive process of building stock price predictive model using ARIMA, and reveals that the ARIMA model has a strong potential for short-term prediction and can compete favourably with existing techniques for stock price prediction. However, stock prices are often highly nonlinear and non-stationary, which limits the long-term prediction applicability of ARIMA model. Compared with traditional machine learning models, Artificial Neural Networks (ANNs) can successfully model complex real-world data by extracting robust features that capture the relevant information and achieve even better performance than before.

The LSTM was first proposed in [10], which is a variant of RNNs. In [2], the LSTM is adopted to predict prices with historical numerical and textual data. [6] simulated a stock trading strategy with the forecast of the LSTM. [13] decomposes the hidden states of memory cells into multiple frequency components to predict the stock prices. [4] presents a deep learning framework where wavelet transforms (WT), stacked autoencoders (SAEs) and long-short term memory (LSTM) are combined for stock price forecasting. The Gated Recurrent Unit (GRU) [5] is a variant of LSTM. [7] introduces a prediction model depend on Bidirectional Gated Recurrent Unit (BGRU) which relies on both online financial news and historical stock prices data to predict the stock movements in the future.

Actually, the short-term prediction is related to recent price fluctuations, while long-term prediction depends more on where the current market is in the stock movement model. Therefore, we use time convolution in conjunction with LSTM to make short and long-term predictions of stock prices by finding long-term operating patterns of stocks.

3 Model Structure

3.1 Time Convolution

Learning the pattern stock fluctuations requires macroscopic considerations. However, LSTM can only process the series data one by one and do not learn the long term pattern of stock. We use 1-d convolution to extract regional features from sequential data, and continuously abstracts the high-level abstract representation of the overall sequence pattern. Therefore, the overall network has fewer parameters and the risk of overfit is even smaller. Through experiments, directly using the convolutional neural network to predict the sequence does have some effect. However, The effect is still unsatisfactory compared to the commonly used recurrent neural network.

3.2 TC-LSTM

The forecast of stock prices is mainly affected by two aspects: one is the short-term price region, and the other is the long-term fluctuation model. We propose a new neural network structure, Time Convolution Long Short-Term Memory (TC-LSTM) network, which uses the LSTM network to simulate the short-term trend of the price series, and learn the long-term sequence pattern through time convolution.

Figure 1 shows the TC-LSTM model being unrolled into a full network, which describes how the value of each gate is updated. The mathematical symbols in Fig. 1 are as follows:

1. x_t is the input vector to the memory cell at time t .
2. w_f , w_s , w_i and w_h are weight matrices.
3. b_f , b_s , b_i and b_h are bias vectors.
4. h_t is the value of the memory cell at time t .

The model is improved from the long-term memory network. It also uses the forget gate layer to control the cell state and the output gate layer to determine the output of the next step. The difference is that a one-dimensional convolution is introduced in the input gate layer to extract the long-term characteristics of the sequence, and the formula for forward propagation of the model is as follows.

Given time series $\{X_t|t = 1, 2 \dots n\}$. The time convolutional layer is defined as a small convolutional neural network, which extracts the overall features of the sequence through convolution and pooling and slides forward with time. Its architecture is shown in Fig. 2. For easier description, we define it as $Conv_t$, and the parameters of the network are defined as V_c . The formula is as Eq. 1.

The forget gate layer uses the sigmoid function, which outputs a threshold between 0 and 1, to determine how much information to keep in the cell state of the previous layer. The forget gate layer is defined as Eq. 2.

$$f_t = Sigmoid(W_f \cdot [Conv_t, X_t, h_{t-1}] + b_f) \quad (1)$$

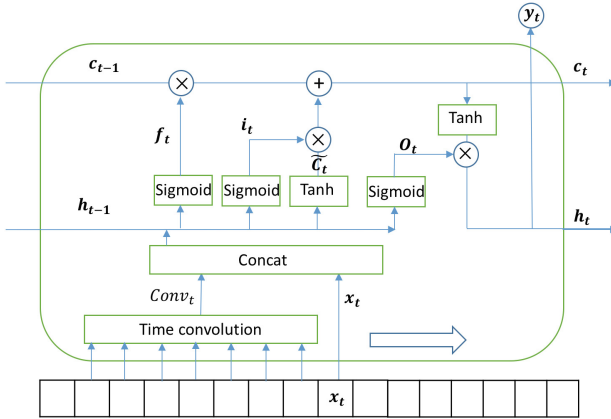


Fig. 1. Architecture of TC-LSTM.

From the old cell state to the new cell state, it is mainly composed of two parts. One is the remaining information in the old cell state after passing through the forget gate layer, and the other is the new data extracted based on the current input information. Through a summation gate circuit integrated, TC-LSTM gets the new cell state c_t . The formula is as Eqs. 3 and 4.

$$s_t = Sigmoid(W_s \cdot [Con_t, X_t, h_{t-1}] + b_s) * tanh(W_i \cdot [Con_t, X_t, h_{t-1}] + b_i) \quad (2)$$

$$c_t = c_{t-1} \cdot f_t + s_t \quad (3)$$

The output of TC-LSTM is jointly controlled by three aspects: the information extracted from the convolution layer, the current data information, and the cell status. The output of the cell is defined as Eq. 5.

$$Con_t = Conv(V_c, [X_p, X_{p+1} \dots X_{t-1}]) \quad (4)$$

$$y_t = h_t = Sigmoid(W_h \cdot [Con_t, X_t, h_{t-1}] + b_h) * tanh(W_o \cdot c_t + b_o) \quad (5)$$

3.3 Price Prediction

The output layer of TC-LSTM is the same as LSTM. Therefore, the output is a vector with the dimension equal to the cell state to get the prediction of price. Consider a time series of stock prices $\{p_t | t = 1, 2 \dots n\}$. Our goal is to make a n-step prediction on p_{t+n} based on the prices up to p_t . Formally, the n-step prediction is defined as Eq. 6.

$$p_{t+n} = f(p_t, p_{t-1}, p_{t-1} \dots p_1) \quad (6)$$

where f denotes the model mapping from the history prices to the price of n-step ahead.

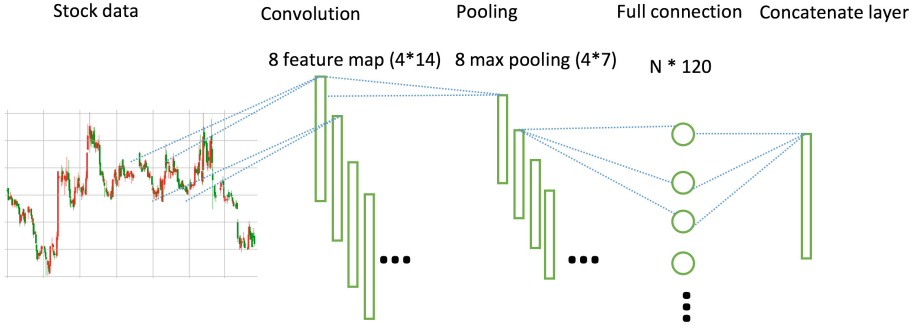


Fig. 2. Architecture of time convolution.

As the scales of prices varies for different stocks, without loss of generality, we normalize the prices $\{p_t|t = 1, 2 \dots n\}$ of each stock to $\{q_t|t = 1, 2 \dots n\}$ on the range $[0, 1]$. The output of the TC-LSTM is a hidden vector h_t that is used for price prediction. Specifically, we apply a matrix transformation on the hidden vector to make the n-step prediction, and the formula is as Eq. 7.

$$v_{t+n} = w_p \cdot h_t + b_p \tag{7}$$

where w_p is a weight vector, and b_p is the bias. Note that although this is a linear transformation, the non-linearity of this price predictor arises from the nonlinear hidden vector h_t .

The architecture of TC-LSTM differs from LSTM that we use time convolution to extract the series feature, so that we can get a big size of the sliding window to train the model. To learn general patterns in the stock market, prices of multiple stocks are used to train the network. Denote the prices as $\{p_t^m|t = 1, 2 \dots T; m = 1, 2 \dots, M\}$ with M stocks, the model is trained by minimizing the mean square error between the predicted and true normalized prices in the training set. The formula of the mean square error (MSR) is as Eq. 8.

$$Loss = \frac{1}{M * T} \sum_{m=1}^M \sum_{t=1}^T (q_{t+n}^m - v_{t+n}^m)^2. \tag{8}$$

All the model parameters are updated through the Backpropagation through time (BPTT) algorithm [11].

4 Experiments

4.1 Data Sets

We used two data sets to test the effectiveness of the model. The first one is the data set used in [13] which collects the daily opening prices of 50 stocks among 10 sectors from 2007 to 2016. The second one is the 50 stocks included

in the SSE 50 Index from 2008 to 2017, which we retrieve from Sina Finance¹. We validate the effectiveness of the model on these two data set to ensure the model's ability to predict on different kind of market.

4.2 Comparison with Other Methods

To test our model, we conduct experiments to compare the proposed TC-LSTM network with the other baseline methods, including LSTM [10], GRU [5], and Naive. In specific, LSTM and GRU are the most widely used recurrent neural network model. Besides, in order to assess the predictive power of the model, we calculate the error of the Naive model as the basic error of the forecast. Naive model uses the latest price as a forecast for the future.

The model implementation uses the Tensorflow framework. In the experiment, we use the MSR between the predicted price and the real price as a standard to test the quality of the model, and the formula of the MSR is as Eq. (8). In TC-LSTM experiment, we set the input sequence length of the time convolution layer to 100 step, so the length of the sequence received by each training of TC-LSTM is $100 + 1$ step. We divide each stock in the data set into 6:2:2 ratios in the time dimension as the train set, test set and validation set. We use the train set to train the model, and select the optimal model by the performance on the test set, and then get the MSR on the validation set. As shown in Table 1, we calculate the MSR of the four models on the validation set with 1, 3, 5, 7, 30, and 60 days prediction.

The results of the US stock market forecast for each model are shown in Table 1, and the results of Shanghai Securities market forecast are shown in Table 2.

Table 1. The MSR of prediction on American stock market data.

	LSTM	GRU	TC-LSTM	Naive
1-day	1.2545	1.2140	0.9770	1.9475
3-day	2.5629	1.5907	1.3958	5.6479
5-day	2.8783	2.2030	2.0321	9.1315
7-day	3.9332	4.2817	3.1685	12.4107
30-day	9.5835	9.3077	7.2412	46.6161
60-day	17.1506	16.8107	12.5943	92.0835

The experimental results on the SSE 50 data are basically consistent with the experimental results of the US stock data. The TC-LSTM model all shows the best performance. It can be seen that the three models have great advantages over the prediction results of Naive model, which indicated that the three models all have a certain predictive ability in the current data set. In terms of the comparison of the three models, the long-term and short-term prediction capabilities

¹ <http://finance.sina.com.cn/stock/>.

Table 2. The MSR of prediction on SSE50.

	LSTM	GRU	TC-LSTM	Naive
1-day	0.03221	0.05115	0.03078	0.19068
3-day	0.12142	0.12076	0.11752	0.58935
5-day	0.19063	0.19802	0.17620	0.98212
7-day	0.26913	0.27826	0.23488	1.34773
30-day	1.20671	1.23302	1.10512	5.48739
60-day	2.64125	2.62933	2.41511	9.67575

of the TC-LSTM model are better than those of the other two models. For example, in predicting the price for 3 days interval, the value of MSR of TC-LSTM is 1.3958 in the American stock market data, which is 45.5% lower than LSTM and 12.3% lower than GRU. Besides, the advantage of the TC-LSTM model is more obvious as the step size increased. From the perspective of the model's predictive effect with step change, as the step size increases, the TC-LSTM model loss rises more smoothly and fluctuates less, and it can be expected that the TC-LSTM performance will become better as the step size continues to increase.

From the experimental results, we can conclude that the TC-LSTM model has two main advantages. The first one is that the overall prediction accuracy is better than other baseline models. The TC-LSTM model adds time convolution, which is equivalent to the whole window sliding acquisition feature in time series with the size of the time convolution. Compared to other baseline models, when the window size is 1 step, the TC-LSTM model obtains more information per training. Thus, it obtains the overall change pattern better. The second one is as the step size increases, from the short-term forecast to the long-term forecast, the TC-LSTM error rises more steadily and the advantage becomes more obvious. It can be seen that the time-convolutional features of the added sequence have certain improvements in the accuracy of the short-term prediction of the sequence, and long-term prediction has a significant effect on the model.

5 Conclusion

This paper considers two factors to predict the stock price: the short-term fluctuations and long-term pattern of the stock. We adapted to these two aspects by introducing time convolution and LSTM, and presented a TC-LSTM network learning fluctuation feature with time convolution to predict the trend of stock prices. The TC-LSTM model is inspired by professionals in the financial field who visualize historical fluctuations to determine trends. We test the MSR of our proposed model compared with the other three baseline models. The results provide evidence that it achieves better performance in both long and short-term prediction.

The TC-LSTM model can be applied to many time series forecasting problems, such as sales forecasting, rainfall forecasting, and air quality index forecast-

ing etc. At present, the experiment is only aimed at more complex and difficult to explain nonlinear stock data. In the future work, we will optimize the model on multiple types of data sets and test the effect of the model on different types of data.






Acknowledgments. This work is supported by the National Key Research and Development Program of China under grants 2016QY01W0202 and 2016TFB0800402, the National Natural Science Foundation of China under grants 61572221, U1401258, 61433006 and 61502185, Guangxi High level innovation Team in Higher Education Institutions–Innovation Team of ASEAN Digital Cloud Big Data Security and Mining Technology.

References

1. Adam, K., Marcet, A., Nicolini, J.P.: Stock market volatility and learning. *J. Fin.* **71**(1), 419–438 (2016)
2. Akita, R., Yoshihara, A., Matsubara, T., Uehara, K.: Deep learning for stock prediction using numerical and textual information. In: IEEE/ACIS International Conference on Computer and Information Science, pp. 1–6 (2016)
3. Ariyo, A.A., Adewumi, A.O., Ayo, C.K.: Stock price prediction using the ARIMA model. In: 2014 UKSim-AMSS 16th International Conference on Computer Modelling and Simulation, pp. 106–112, March 2014. <https://doi.org/10.1109/UKSim.2014.67>
4. Bao, W., Yue, J., Rao, Y.: A deep learning framework for financial time series using stacked autoencoders and long-short term memory. *PLoS ONE* **12**(7), e0180944 (2017)
5. Cho, K., van Merriënboer, B., Bahdanau, D., Bengio, Y.: On the properties of neural machine translation: encoder-decoder approaches. *CoRR* abs/1409.1259 (2014). <http://arxiv.org/abs/1409.1259>
6. Gao, Q.: Stock market forecasting using recurrent neural network. Ph.D. thesis, University of Missouri-Columbia
7. Huynh, H.D., Dang, L.M., Duong, D.: A new model for stock price movements prediction using deep neural network. In: The Eighth International Symposium, pp. 57–62 (2017)
8. Kolen, J.F., Kremer, S.C.: Gradient flow in recurrent nets: the difficulty of learning longterm dependencies. *Field Guide Dyn. Recurr. Neural Netw.* **28**(2), 237–243 (2001)
9. Patel, J., Shah, S., Thakkar, P., Kotecha, K.: Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques. *Expert Syst. Appl. Int. J.* **42**(1), 259–268 (2015)
10. Hochreiter, S., Schmidhuber, J.: Long short-term memory. *Neural Comput.* **9**(8), 1735–1780 (1997)
11. Werbos, P.J.: Backpropagation through time: what it does and how to do it. *Proc. IEEE* **78**(10), 1550–1560 (1990)
12. Xiao, Y., Xiao, J., Liu, J., Wang, S.: A multiscale modeling approach incorporating ARIMA and anns for financial market volatility forecasting. *J. Syst. Sci. Complex.* **27**(1), 225–236 (2014)
13. Zhang, L., Aggarwal, C., Qi, G.J.: Stock price prediction via discovering multi-frequency trading patterns. In: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 2141–2149 (2017)



Web Data Extraction from Scientific Publishers' Website Using Hidden Markov Model

Jing Huang^{1,2} , Ziyu Liu^{1,2} , Beibei Wang^{1,2} ,
Mingyue Duan^{1,2} , and Bo Yang^{1,2(✉)} 

¹ College of Computer Science and Technology, Jilin University,
Changchun 130012, China

ybo@jlu.edu.cn

² Key Laboratory of Symbol Computation and Knowledge Engineering,
Jilin University, Ministry of Education, Changchun 130012, China

Abstract. Recently, large amounts of information on web pages have been emerging in an endless stream. And numerous papers are published on more than three thousands of journals, especially in the field of technology. It's almost impossible for the user to search the information one by one. The user has to click a lot of links when he or she wants to get information among the thousands of journals, such as the introduction of the journals, impact factor, ISSN and so on. To solve this problem, it's necessary to develop an automatic method that filter the information out of deep web automatically. The method in this paper is able to help people quickly get needed information classified and extracted. This paper contains the following work: firstly, the method of machine learning, HMM, is used to extract the journal information from the publisher's website, which improves the generalization ability of using the heuristic method; then, during the data processing step, content extraction technique is used to improve the performance of Hidden Markov Model; finally, we store the extracted information in a structured way and display it. In the experimental step, three algorithms are tested and compared in the accuracy, recall and F-measure, the results show that HMM with content extraction (C-HMM) has the best performance.

Keywords: Web information extraction · Hidden markov model
Content extraction

1 Introduction

With the rapid development of Internet technology, the data in the network has been growing exponentially. Web pages are constructed by HTML, CSS styles, and AJAX. And pages are prettified by different scripting languages. As a result, web pages become different from each other. Normally, search engines are used to retrieve information. However, web pages that cannot be indexed by search engines are 400 to 500 times than the number of surface webs. This phenomenon is showed in [1]. Therefore, if some details are needed, a series of hyperlinks must be clicked on.

Traditional web data extraction is a process of extracting structured information from unstructured data. And the generalized data extraction process is shown in Fig. 1.

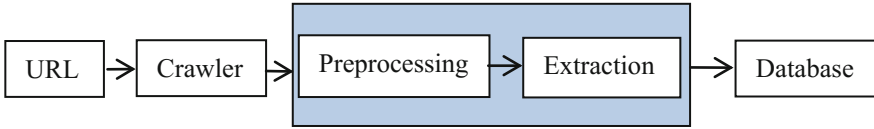


Fig. 1. The generalized data extraction process.

At present, the methods and technology about this hot research topic are: (1) The wrapper technology is suitable for fixed-format web pages [2], but it is difficult to transplant to other pages and maintain; (2) The information extraction method based on NLP is suitable for the extraction task of plain text. But in term of web data, it cannot be directly used by label segmentation; (3) Ontology-based information extraction methods require large cost to construct the ontology [3, 4]; (4) Technology using the DOM tree is based on the structure of the web page, which is suitable for web pages with similar structure, including DSE algorithm [5] and MDR algorithm [6]; (5) The accurate of heuristic-based web information extraction algorithm [7] is relatively high, but the scope of its application is limited. The hidden Markov model (HMM) used in this paper has many advantages contrast to the above methods, such as easy to establish, easy to transplant and so on.

The paper is organized as follow. Section 2 describes the related works including the issues and the definition of HMM, Sect. 3 explains the proposed framework, Sect. 4 represents the experimental results and analyze it, Sect. 5 concludes the paper and clarifies the future work.

2 Related Works

2.1 The Issues

HMM, a finite state of probability, is shown in Fig. 2. It contains the observation layer and hidden layer [10]. The observation layer is the observation sequence that to be recognized. And the hidden layer is the markov chain with the probability of state transition.

Traditional HMM is regarded as the most successful model in the field of speech recognition, but it is not so perfect to deal with the semi-structured web information.

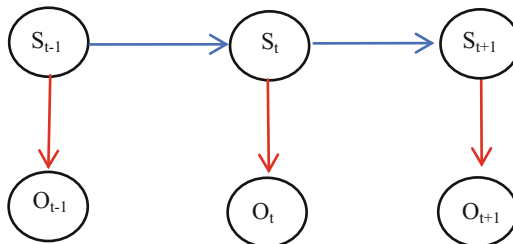


Fig. 2. Hidden Markov chain.

According to [8], because most HMM based approaches take single term as basic unit for information extraction. However web pages are usually consists of multiple content blocks. In these blocks, logically related contents are grouped together. This clustering property of web content provides additional information for improving web data extraction. So the semantic blocks are regarded as the observation sequence.

2.2 Definition of HMM

The modified definition of HMM is shown as follows:

- N = number of the states in the model
- M = number of the observation symbols, which are the semantic blocks
- $S = \{S_1, S_2, \dots, S_N\}$, the states set
- $O = \{O_1, O_2, \dots, O_M\}$, the observations (semantic blocks) set
- $\pi = \{\pi_1, \pi_2, \dots, \pi_N\}$, the initial state distribution set
- $\mathbf{A} = \{a_{ij}\}$ the state transition probability distribution, where a_{ij} is the probability of the transfer from S_i to S_j
- $\mathbf{B} = \{b_j(k)\}$, the emission probability distribution at state S_j .

From the beginning state, a HMM transitions from one state to another and produces an output string $\{o_1, o_2, \dots, o_k\}$, until the end state is reached. In general, an output string can be generated by many paths and has its own probabilities. The sum of these probabilities for each path is the full probability of producing this output string. Thus, the symbols are taken from the discrete dictionary and the HMM produces a probability distribution. Training data helps to learn this distribution. During testing, the HMM produces the most likely state transitions to produce the output string.

HMM can be represented as $\lambda = \{\pi, \mathbf{A}, \mathbf{B}\}$ after getting the parameters. For a given set of observations, corresponding state set can be get by Viterbi algorithm [9].

3 Journal Information Extraction Using HMM

The framework of journal information extraction is shown in Fig. 3. First, URL seeds are crawled from web and divided them into two parts, training web page and testing web page. Then through the processing, web page is transformed into semantic blocks. After learning the parameters of HMM and testing it, the labeled blocks are stored into database.

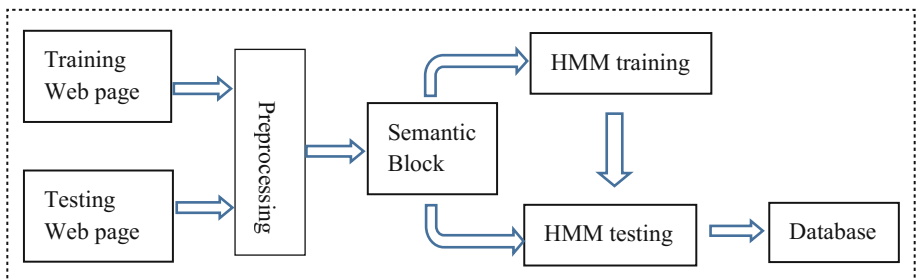


Fig. 3. Journal information extraction framework.

3.1 Website Information Crawling

If a set of journals' names are given. Then a series of hyperlinks will be gotten by searching the name of the journal from the search engine. According to some keywords, the hyperlinks can be determined to be the journal' homepages or not. If the page meets the requirements, add the hyperlinks to the URL seeds.

3.2 Content Extraction

There are both relevant information and irrelevant information in a Web page, the irrelevant information brings some negative influence to HMM. In order to reduce this kind of influence, the irrelevant parts are deleted from pages [11]. Therefore, before using HMM to extract information from Web data, preprocessing is needed.

The first preprocessing is to ignore web page tags and code that are not related to the body. According to the research and analysis of HTML tags, as is shown in Table 1, the following types of tags have little relationship with the topic part of the web page. So they can be filtered first in order to facilitate the next step.

Table 1. Web page preprocessing

Type	Regular expression	Processing method
Web page title	<title>[\s\S]*?</title>	Keep
Script	<script[^>]*?>[\s\S]*?</script>	Delete
NoScript	<noscript[^>]*?>[\s\S]*?</noscript>	Delete
CSS style	<style[^>]*?>[\s\S]*?</style>	Delete
Annotation	<!--[\s\S]*?-->	Delete
Blank line	[\s\r\n\t]+	Delete

The next content extraction process is divided into the following two steps.

Split Into Blocks: in this paper, the method proposed in [12] to divide the web page into blocks is used. The criterion for the partition is the <div> tag. The web page is divided into blocks, which can avoid the influence of the non-standard web page on building the DOM tree. And the blocks can quickly and accurately partition the web page. What's more, the process to split the web pages data into blocks is simple.

Choose Blocks: The web page structure that contains nested relationships can be shown in block sequence without nested relationships. By analyzing each block, appropriate blocks can be chosen.

Definition of text block. The block which text density is more than threshold $p = 0.7$ and the number of punctuation marks is more than $q = 40$ is call test block. The text block mainly exists in the main part of the web page.

Definition of link block. The link block is that mainly exist in the form of hyperlinks. Its text density is small. Or the block does not contain any comma or period. Most of the link blocks appear in the "noise" area.

The method of choosing blocks is as follows: (1) Find the text block which has the most text; (2) Search for the first link blocks which continuously appear before the text

block. (3) Search for the first contiguous link blocks that continuously appear after the text block. (4) The content between the link blocks obtained from the (3) and (4) is regarded as the main content of the current web page.

3.3 Sequence Labeling

In machine learning, HMM mainly deal with two problems in extracting information. Learning in training and decoding in the extraction process. HMM can be regarded as $\lambda = \{\pi, \mathbf{A}, \mathbf{B}\}$, and the purpose of the learning process is aimed at λ .

In this paper, Maximum Likelihood Algorithm is used. Because the greatest advantage of this algorithm is that through marking, we can find out the random process from the hidden state transitions. We no longer need to adjust the parameters of HMM through experiment, but can calculate them by taking statistics.

The formula of calculating model parameters is as following:

$$\pi_i = \frac{Init(i)}{\sum_{j=1}^N Init(j)}, \quad 1 \leq i \leq N \quad (1)$$

Where $Init(i)$ is the number of sequences which start from state i in all training sequences. π_i is the probability that the HMM chain will start in state i .

$$a_{ij} = \frac{C_{ij}}{\sum_{k=1}^N C_{ik}}, \quad 1 \leq i, j \leq N \quad (2)$$

For each a_{ij} representing the probability of moving from state i to state j . Where C_{ij} is the number of transitions from state i to state j in all training sequences. During this process, if there is are continuous redundant blocks, we only keep two blocks in the following formula.

$$b_j(k) = \frac{E_j(k)}{\sum_{i=1}^M E_j(i)}, \quad 1 \leq j \leq N, 1 \leq k \leq M \quad (3)$$

For each expressing representing the probability of an observation k be generated from state j . Where $E_j(w)$ shows how many times block with Symbol w is observed at state j in all training sequences.

3.4 Storage and Display

Finally, the extracted information is stored in structured form in database and display it. In this paper, MySQL is used to store the journals' information. It will be displayed in our APP [16].

4 Experiment

The experiment was based on the data of Elsevier [13], Springer [14] and Wiley [15] publishers’ web sites, as is shown in Table 2. Y represents the existence of this attribute, and N is opposite. In this paper, journals’ information including introduction, ISSN and impact factor are extracted from publishers’ web sites. Due to the Wiley doesn’t have impact factor and ISSN in journal’s homepage, we just extract the journal’s introduction from Wiley.

Table 2. Dataset statistics

	Introduction	ISSN	Impact factor	The number of training URL	The number of testing URL
Elsevier	Y	Y	Y	43	363
Springer	Y	Y	Y	28	188
Wiley	Y	N	N	29	195

The proposed module fetches the structured data embedded in unstructured HTML pages. Then stores it into database. The extraction results are shown in the Table 3. The three classic metrics, precision, recall and F-Measure of information extraction based on HMM are better than contrast algorithm

Heuristics VS C-HMM demonstrates that C-HMM has a better performance than data extraction based on heuristics. Now analyze the reason why this improved

Table 3. Heuristics VS HMM VS C-HMM

	Heuristics			HMM			C-HMM		
	P	R	F1	P	R	F1	P	R	F1
<i>Elsevier</i>									
Introduction	0.977	0.977	0.977	0.908	0.908	0.908	1	1	1
ISSN	0.977	1	0.988	1	1	1	1	1	1
Impact Factor	0.977	1	0.988	1	1	1	1	1	1
<i>Springer</i>									
Introduction	0.968	0.968	0.968	0.921	0.921	0.921	0.974	0.974	0.974
ISSN	0.968	0.968	0.968	0.500	1	0.667	1	1	1
Impact factor	0.968	0.968	0.968	1	1	1	1	1	1
<i>Wiley</i>									
Introduction	1	1	1	0.964	0.964	0.964	1	1	1
<i>Total</i>									
Introduction	×	×	×	0.921	0.921	0.921	0.994	0.994	0.994
ISSN	×	×	×	0.746	1	0.855	1	1	1
Impact factor	×	×	×	1	1	1	1	1	1

performance appears. The data extraction based on heuristics is to use different templates to extract the information from different publishers' web sites, for a web site, it has only a template to work. "×" in the Table 3 represent that it can not form a unified model to calculate the three classic metrics. So the data extraction based on heuristics can not deal with the web sites which contains differently structured web pages. The proposed framework in this paper trains the model through different web sites. Therefore, the unified model has a better ability of generalization.

HMM VS C-HMM demonstrates framework in this paper has a better performance than the same method without content extraction (Sect. 3.2). The reason why this improved performance appears is that web pages without content extraction will contains many "noise" blocks (navigation bar, related links, advertising links, copyright notices, etc.). The "noise" blocks may be regarded as introduction, ISSN or impact factor during the information extraction process. The proposed framework in this paper reduce the "noise" blocks, therefore, the framework has a better rate of accuracy.

5 Conclusion and Future Work

As an important branch of NLP, the information extraction is a deeper data mining process than information retrieval, and its research value is more and more important in the era of massive information. We proposed a framework that use HMM with content extraction to extract journals' information from publishers' web sites. This method has a better extracting performance than heuristic algorithm and HMM without content extraction. And improve the generalization ability of the model than heuristic algorithm. In the future, we will focus on other machine learning and deep learning techniques in order to improve the extracting performance further.

Acknowledgments. This work was supported in part by National Natural Science Foundation of China under grants 61373053 and 61572226, and Jilin Province Key Scientific and Technological Research and Development project under grants 20180201044GX and 20180201067GX.

References

1. Bergman, M.: The deep web: surfacing hidden value. *J. Electron. Publ.* **7**(1), 1–14 (2001)
2. Crescenzi, V., Mecca, G., Meriardo, P.: RoadRunner: towards automatic data extraction from large web sites. In: *27th International Conference on Very Large Data Bases*, pp. 109–118. Morgan Kaufmann, Roma, Italy (2001)
3. Gutierrez, F., Dou, D., Fickas, S., et al.: A hybrid ontology-based information extraction system. *J. Inf. Sci.* **42**(6), 798–820 (2016)
4. Zhang, N., Chen, H., Wang, Y., et al.: Odaies: ontology-driven adaptive Web information extraction system. In: *IEEE/WIC International Conference on Intelligent Agent Technology*, pp. 454–460. IEEE (2003)
5. Wang, J., Lochovsky, F.H.: Data-rich section extraction from HTML pages. In: *International Conference on Web Information Systems Engineering*, pp. 313–322. IEEE, Singapore (2003)

6. Liu, B., Grossman, R., Zhai, Y.: Mining data records in Web pages. In: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 601–606. ACM (2003)
7. Kumaresan, U., Ramanujam, K.: Web data extraction from scientific publishers' website using heuristic algorithm. *Int. J. Intell. Syst. Appl.* **9**(10), 31–39 (2017)
8. Zhong, P., Chen, J.: A generalized hidden markov model approach for web information extraction. In: IEEE/WIC/ACM International Conference on Web Intelligence, pp. 709–718. IEEE, Hong Kong (2006)
9. Forney, G.: The Viterbi algorithm. *Proc. IEEE* **61**(3), 268–278 (1973)
10. Rabiner, L.R., Juang, B.H.: An introduction to hidden Markov models. *IEEE ASSP Mag.* **3**(1), 4–16 (1986)
11. Lai, J., Liu, Q., Liu, Y.: Web information extraction based on hidden Markov model. In: 14th International Conference on Computer Supported Cooperative Work in Design, pp. 234–238. IEEE, Shanghai (2010)
12. Xiong, Z., Lin, X., Zhang, Y., Ya, M.: Content extraction method combining web page structure and text feature. *Comput. Eng.* **39**(12), 200–203 (2013)
13. Elsevier. <https://www.elsevier.com/>. Accessed 25 Apr 2018
14. Springer. <https://link.springer.com/>. Accessed 25 Apr 2018
15. Wiley. <https://onlinelibrary.wiley.com/>. Accessed 25 Apr 2018
16. APP download link. <http://www.acheadline.com/>

Knowledge Graph and Knowledge Management



MedSim: A Novel Semantic Similarity Measure in Bio-medical Knowledge Graphs

Kai Lei¹, Kaiqi Yuan¹, Qiang Zhang¹, and Ying Shen¹✉

School of Electronic and Computer Engineering,
Peking University Shenzhen Graduate School, Shenzhen, China
{leik,shenyng}@pkusz.edu.cn, kqyuan@pku.edu.cn,
zhangqiang@sz.pku.edu.cn

Abstract. We present MedSim, a novel semantic **SIM**ilarity method based on public well-established bio-MEDical knowledge graphs (KGs) and large-scale corpus, to study the therapeutic substitution of antibiotics. Besides hierarchy and corpus of KGs, MedSim further interprets medicine characteristics by constructing multi-dimensional medicine-specific feature vectors. Dataset of 528 antibiotic pairs scored by doctors is applied for evaluation and MedSim has produced statistically significant improvement over other semantic similarity methods. Furthermore, some promising applications of MedSim in drug substitution and drug abuse prevention are presented in case study.

Keywords: Semantic similarity · Semantic networks · Bioinformatics

1 Introduction

Semantic similarity metric is widely used in medical information retrieval [1] and medical knowledge reasoning. The most promising application scenario is therapeutic substitution, also known as therapeutic interchange and drug substitution. It is the practice of replacing one prescription with chemically different drugs that are expected to have the same clinical effect. Medicine semantic similarity measure plays an important role in this context by enabling a proper interpretation of drug information [2].

Unlike conventional similarity measures, semantic similarity methods based on Knowledge Graph (KG) have been proven effective in nature language processing and information retrieval [3]. KG is a type of graph structure that records massive entities and relations, such as FreeBase [4] and DrugBank [5]. DrugBank is a well-known bioinformatics KG for its broad scope and great integrity, which classifies medicines with multi-category bio-medical knowledge bases.

Supported by the National Natural Science Foundation of China (No. 61602013) and the Shenzhen Key Fundamental Research Projects (Grant No. JCYJ20170818091546869 and JCYJ20170412151008290).

The existing KG-based semantic similarity can be classified into structure-based similarity measures, and corpus-based ones. Aimed at heterogeneous networks, PathSim [6], and random walk [7] are introduced to compute semantic similarity among the same type of objects by fully utilizing the path information. IC-based measures primarily rely on the contextual information of words, which usually measures the general semantic relevance between two words [8]. Distributed representation methods [9] calculate semantic similarity by transforming concepts into dense low-dimensional vectors learned from the large scale of corpus. Hybrid measures combine structural features with corpus features to overcome data sparseness and data noise. A weighted path method (Wpath) [10] is proposed by employing both path length and information content. SimCat [11] incorporates category corpus and relationship structure information. However, due to the limited coverage of KGs, the aforementioned measures cannot be directly applied to a specific domain.

The bio-medical field has complex and diverse terminology, hierarchies and attributes to be considered. Pedersen et al. [12] presented a cluster-based approach with new features and evaluated this method for two different bioinformatic knowledge bases within the UMLS. Traverso et al. [13] proposed GADES to compare entities in bioinformatic knowledge graphs by encoding the KG in aspects, e.g., hierarchies, neighborhoods, and specificity. Even though these semantic similarity methods consider the characteristics in the bio-medical field, they often rely on a limited number of data sources and are validated in a limited scale of dataset. Besides, they exclusively depend on the KG-mined features rather than fully utilized the textual information of KG.

To address the problem above, we propose MedSim, a novel semantic similarity method based on public well-established medicine KG and multi-category data sources, to study the therapeutic substitution of antibiotics. Antibiotics are extensively applied as antimicrobial agents in disease treatment but the abuse of antibiotics is becoming increasingly serious. We consider the semantic similarity between two antibiotics in bio-medical category using not only medicine-specific features but also the structure and corpus information of DrugBank which is freely accessible. To our knowledge, the biomedical domain never witnesses the standard human rating datasets for semantic similarity publications. A dataset labeled by doctors which is much larger than other medical similarity methods [2, 14] is applied to evaluate MedSim. To make our method more reproducible, the dataset is freely accessible in Github¹. The main contributions of this work can be summarized as followed:

- To improve the interpretability of drug property and the context-based word representation, the one-dimensional vector of medicine-specific features is transformed to multi-dimensional weighted vectors. The medicine-specific features reflect medicine characteristics and can be extended to all types of medicine.
- To reduce the noises introduced by hierarchical structure on semantic similarity metrics, we employ a KG-based hierarchy embedding feature and

¹ <https://github.com/YuanKQ/MedSim-antibiotics-labeled-dataset>.

corpus-based semantic-level features, of which the combination can mine more information of interest and simplify the labor-intensive process compared with universal fields.

- Experiment results show that compared with the existing methods, MedSim can evaluate similarity more effectively. This reveals that on the analytics and assessments of semantic network, domain specific features, structural and textual information are important.

The rest of this paper is organized as follows: Sect. 2 presents the process of dataset. Section 3 proposes our semantic similarity method MedSim based on bio-medical KGs. Section 4 reports the evaluation experiments and explains the evaluation results. Conclusions and future work are outlined in Sect. 5.

2 Data Processing

2.1 Preparation of Data Source

Semantic similarity measures relied on single data source provide only partial information about a subset of interest and the computed results show various degree of incompleteness. To address this problem, MedSim integrates the following well-established and widely used multi-category data sources.

DrugBank. DrugBank [5] is a comprehensive bioinformatics and cheminformatics KG that combines detailed drug entities with drug information. It contains 10,513 drug entities including 1,739 approved small molecule drugs, 873 approved biotech (protein/peptide) drugs, 105 nutraceuticals and over 5,029 experimental drugs. Each drug entity contains more than 200 properties, such as chemical structure, prescription, pharmacology, pharmacoeconomics, spectra, etc.

SIDER. SIDER [15] is a side effect database of information on marketed medicines and their recorded adverse drug reactions, including 1430 drugs and 5868 side effects. The relationships between antibiotics and the corresponding side effects will be extracted to calculate the side effect based similarity in MedSim.

NDF-RT. National Drug File - Reference Terminology (NDF-RT) [16] combines the hierarchical drug classification with multiple drug characteristics including physiologic effect, mechanism of action, pharmacokinetics, etc. We extract mechanism of the essential pharmacologic properties of medications (physiologic effect and mechanism of action) from NDF-RT.

PubMed. PubMed [17] is a bio-medical search engine accessing more than 27 million citations for biomedical literature from MEDLINE, life science journals, and online books. We crawl more than 500,000 papers about medicine via the PubMed API to help establishing the semantic features of MedSim.

2.2 Antibiotic Pairs Labeling

To verify the effectiveness of MedSim, we conduct experiments on 52 most commonly used antibiotics of 10 categories in hospital. With the combination of these antibiotics, 1326 pairs are generated. 528 randomly selected pairs cover nearly 40% of the total. Referring to [18,19], doctors, from the perspective of clinical application, score the similarity between two antibiotics, which ranges in $[0, 1]$, according to both antibacterial spectrum and efficacies of medicine. 0 indicates that there is no similarity between two antibiotics, while 1 implies that the two antibiotics are extremely similar. The adverse reactions, side effects, patient’s past history and other factors are left aside in this stage. To make antibiotic pairs labeling more accurate, each pair is labeled by at least 3 doctors and the average is taken as final result. The Pearson coefficient between the scores issued by each doctor and the average score ranges from 82.7% to 86.4% while Spearman coefficient ranges from 79.2% to 88.8%, both proving the reliability of doctors’ assessment. Scores about the antibiotic similarity were uploaded to Github. The labeled antibiotic pairs are divided into training set and test set, which will be used in our regression prediction model in Sect. 4.

3 Methodology

MedSim is a medicine similarity metric predicted by the random forest regression model learned from the following features: medicine-specific features (side effect, target protein, mechanism of action, physiological effect), structure feature of concept taxonomy (hierarchy embedding-based feature), and semantic-level features (KG-based semantic textual similarity and word embedding-based feature).

3.1 Medicine-Specific Features (MF)

Domain-specific KGs and multi-category corpus are adopted to mine medical-specific features, so as to address the incompleteness of drug attributes from single data source. Side effect, target protein, mechanism of action, and physiological effect are utilized to explore the medicine-specific features which can simplify the semantic representation of medicines.

Instead of simply flattening all properties into one vector, the weighted property vectors of different features from multiple data sources are generated to interpret the characteristic of drugs. For a drug, its multi-dimensional weighted feature vector is erected by stacking weighted vectors of all medicine-specific features. Each row of vector represents one category of characteristics, in which the values demonstrate the weights of specific properties. Figure 1 shows a snapshot of multi-dimensional weighted feature vector of an antibiotic: nitrofurantoin.

Side Effect Based Similarity. For a drug d , its side effects can be obtained from SIDER database. In the paper, we want to find out side effects related to some drug, as well as those specific to this certain drug, hence improve the

Side Effect	0	0	0	...	0	0.545	...
Target	0	0	0	...	7.074	0	...
Mechanism	0	3.640	0	...	3.855	0	...
Physiology	7.871	0	0	...	0	0	...

Fig. 1. Part of multi-dimensional weighted feature vector of nitrofurantoin.

discrimination. Inverse Document Frequency (IDF) can work well in alleviating the impact of high frequency terms and pay more attention to rare ones:

$$IDF(s, Drugs) = \frac{\log(|Drugs| + 1)}{DF(s, Drugs) + 1} \quad (1)$$

Where $Drugs$ is the set of all drugs, s is a side effect, and $DF(s, Drugs)$ is the number of drugs with the side effect s . The weighted side effect vector of a drug d is $sider(d)$, consisting of side effects extracted from SIDER. The value of element s of $sider(d)$, denoted $sider(d)[t]$, is $IDF(s, Drugs)$ if it is one of the side effects of drug d , otherwise it is 0. The side effect-based similarity of two drugs d_1, d_2 is the cosine distance of the vectors $sider(d_1)$ and $sider(d_2)$.

Target Based Similarity. The information about proteins targeted by a drug d is collected from DrugBank. The target-based similarity of two drugs d_1, d_2 is defined as the cosine similarity of IDF-weighted target protein vectors of two drugs, which are calculated like the IDF-weighted side effect vector.

Mechanism Based Similarity. We collect all the mechanisms of a drug from NDF-RT. The mechanism-based similarity of two drugs is calculated by the cosine distance of IDF-weighted mechanism vectors of two drugs as mentioned in the previous paragraph.

Physiological Effect Based Similarity. The IDF-weighted physiological effect vectors of drugs are also established from NDF-RT, and physiological effect-based similarity measure is the same as the mechanism-based similarity.

3.2 Hierarchy Embedding-Based Feature (HF)

To fully take advantage of taxonomy hierarchy information, DeepWalk [20] is applied to learn the hierarchy embedding from the taxonomy hierarchy in DrugBank. All concepts in taxonomy hierarchy are actually arranged as a concept graph, where nodes represent concepts and edges indicate hierarchical relations.

Unlike some structure-based methods which focus on concrete properties of KG, e.g. neighbors, class hierarchies and node degrees, hierarchy embedding based feature is able to map a total knowledge graph into a low dimension vector space while preserving certain properties of the original graph. Due to

the limited depth of taxonomy hierarchy, the training paths in DeepWalk can cover both complete leaf to root path and neighbor nodes, which can fully utilize structural information. The hierarchy embedding-based similarity is calculated by the cosine distance between two drug vectors.

3.3 Semantic-Level Features (SF)

We adopt two types of semantic-level features: the KG-based semantic textual similarity feature is used to process the entity related textual information, while the word embedding-based similarity feature learned from the context of drugs is used to cluster similar entities in vector space.

KG-Based Semantic Textual Similarity (KSTS). Traditional IC-based semantic similarity metrics require a large domain corpus and cost-intensive labor to remove redundant data. KGs have already mined topic-related knowledge from textual corpus, which has prepared a high-quality domain corpus. The entity description or other textual information about the concepts in KG usually implies the nature of the concepts. The greater is the similarity among the concepts, the greater is the similarity of the words in their entity description.

The bio-medical proper nouns in the entity description of universal KG tend to have few information, which cannot improve the performance of the semantic textual similarity measures [21]. In this context, BM25 algorithm [22] is applied to compute the textual similarity based on entity description by converting the ranking score into the similarity score that ranges between 0 and 1.

Given a description of a drug d_1 containing the keywords q_1, q_2, \dots, q_n , the BM25 score of a description D of another drug d_2 , is

$$score(d_1, d_2) = \sum_{i=1}^n IDF(q_i) \cdot \frac{f(q_i, D) \cdot (k_1 + 1)}{f(q_i, D) + k_1 \cdot (1 - b + b \cdot \frac{|D|}{avgdl})} \quad (2)$$

where $IDF(q_i)$ is the IDF weight of the keywords q_i , $f(q_i, D)$ is the occurrence frequency in the description D , $|D|$ is the number of words in D , and $avgdl$ is the average number of words of all entity descriptions drawn from DrugBank. Usually, the free parameters $k_1 \in [1.2, 2.0]$ and $b = 0.75$. Here, we set $k_1 = 2$ and $b = 0.75$. To normalize the BM25 ranking score, the KG-based Semantic Textual Similarity is defined below, where $Drugs$ is the set of all drugs:

$$KSTS(d_1, d_2) = \frac{score(d_1, d_2) - \min\{score(x, y) | x, y \in Drugs\}}{\max\{score(x, y) | x, y \in Drugs\} - \min\{score(x, y) | x, y \in Drugs\}} \quad (3)$$

The entity description included in DrugBank is few in words but large in numbers. Thereby, BM25 can quickly measure the textual semantic similarity between a drug and the rest.

Textual Embedding-Based Similarity. Word2vec is applied to train the textual embedding vector of PubMed 500,000 indexed papers and medical corpus (e.g. DrugBank and DailyMed). We use the skip-gram model rather than CBOV according to the pre-experimental results. Since word2vec can predict concept relatedness by simple algebraic operations in vector space, the word embedding-based similarity is calculated by the cosine similarity between vectors of corresponding drugs.

3.4 Random Forest Regression Model

Based on the aforementioned features, random forest regression model is applied to measure the medicines semantic similarity. Random forest is an effective ensemble learning algorithm for regression task. In this paper, random forest is used to predict the similarity of an antibiotic pair (a scalar dependent variable y) learned from the selected features of samples (explanatory variables X).

Ten-fold cross validation is applied to train the model. The training and test datasets are from the labeled antibiotic pairs mentioned in Sect. 3.2, except some pairs whose labeled scores substantially differ. The Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Pearson correlation coefficient and Spearman rank correlation coefficient are adopted to evaluate the model. To select the best regression model, we make a detailed comparison of the random forest regression model and other common methods such as linear regression, logistic regression, polynomial regression, tree regression, etc.

As shown in Table 1, the model that we employed has the lowest RMSE and MAE, which indicates that it outperforms other models in precision and stability. The Pearson and Spearman coefficient specify that the similarity measured by the random forest has a strong correlation with the results scored by doctors. Compared to other regression models, random forest is not very sensitive to missing data, which alleviates the impact from the incompleteness of drug attributes. Besides, the randomized sampling before bagging and the application of averaging can avoid overfitting and further improve the generalization ability.

Table 1. Performance comparison of regression models.

Regression model	Pearson	Spearman	RMSE	MAE
Logistic regression	0.273	0.239	2.391	1.641
Linear regression	0.306	0.232	0.186	0.137
Polynomial regression	0.431	0.442	0.178	0.137
Support vector regression	0.456	0.435	0.175	0.124
Adaboost regression	0.465	0.435	0.172	0.129
Bagging regression	0.520	0.474	0.164	0.123
Tree regression	0.563	0.510	0.161	0.116
Random forest regression	0.584	0.518	0.156	0.116

4 Evaluation

To assess the performance of our model, we conduct two types of experiments. We first compare it with the state-of-art semantic similarity methods to prove whether MedSim outperforms others. Furthermore, we measure the prediction ability of individual features and different feature combinations to exploit the effect of the model performance. Spearman and Pearson correlations coefficients are widely used to evaluate semantic similarity measures. In this section, both coefficients are adopted to evaluate the correlation between doctor assessment and experiment results, while Z-significance test between MedSim and baselines is used to evaluate whether MedSim statistically outperforms other methods using two-sided test and 0.05 statistical significance.

4.1 Comparison with State-of-Art Similarity Metrics

We compare MedSim with four state-of-art algorithms, including GADES [13], Res [8], Wpath [10] and Hybrids [14]. The GADES is a structure-based measure, while Res is an information content based measure. Wpath considers both path and IC information. Based on Wpath, the method Hybrids takes medical properties into account to calculate the drug similarity.

As shown in Table 2, different semantic similarity method has different level of correlation between doctor’s judgment and MedSim outperforms the others.

Table 2. Comparison between semantic method.

Method	Pearson	Spearman	Z significance test	
			Z statistic	p-value
GADES	0.251	0.203	2.881	0.051
Res	0.211	0.223	0.273	0.000
Wpath	0.251	0.205	0.805	0.000
Hybrids	0.256	0.278	0.995	0.000
MedSim	0.586	0.523	N/A	N/A

GADES has the lowest Spearman correlation, probably due to the structure of bio-medical KGs. It is common sense that in KGs, the upper-level concepts in a taxonomy are supposed to be more general hence have more entities. However, it may be different in bio-medical KGs, where the entity number of lower-level concepts would be larger than that of upper-level concepts. For example, according to concept tree in the latest released version (version 5.0.6) of DrugBank, the level of Tetracyclines is upper than that of Aminobenzene sulfonamides. However, the entity number of the former one is 3246, which is far less than that of the latter one, which has 235515 entities. Thereby, structure based approach like GADES cannot work well in the bio-medical KG-based similarity measures.

Res also has the lowest Pearson correlation, which also implies the limited effect of IC-based measures in computing medicine semantic similarity.

Wpath shows a slightly improvement over GADES and Res by adopting both structure information and semantic information of KGs. When we set Wpath's free parameter $k = 0.85$, Wpath can achieve its own highest correlation score.

The Hybrids method takes all aforementioned medicine-specific features into account to measure the semantic similarity. Its highest score among all baselines indicates the significance of the medicine-specific features.

Both Pearson and Spearman coefficient of MedSim are over 0.5, indicating that the prediction of our model has a high correlation with doctors' judgment. Compared with other methods, MedSim can more effectively evaluate similarity. The results of the z-test also show that MedSim has a statistically significant improvement over baselines, since in each baseline z statistics are larger than 1.96 and p-values are below the significance level of 0.05. Experiment results also reveal that on the analytics and assessments of KG semantic/structure information, domain specific features need to be considered simultaneously.

4.2 Feature Selection Comparison

The prediction ability of each feature and feature combinations is measured in this section (Table 3).

Table 3. Comparison of feature performance.

	Feature	Pearson	Spearman
Single feature	MF	0.407	0.389
	HF	0.159	0.150
	SF	0.339	0.258
Multiple feature	MF and HF	0.551	0.489
	MF and SF	0.570	0.515
	MF, HF and SF	0.585	0.523

For the single feature, the coefficient scores indicate that medicine-specific features yield a good performance without cooperating with other features.

The combinations of MF and HF and the combinations of MF and SF generally show much better performance than using these features separately, increasing the coefficients by at least 10%. The best performance is obtained by the combination of all features, indicating that the proper combination of features can mine more information and improve the prediction performance.

There are four types of medicine-specific features adopted in our study, among which, the physiological effect based similarity with 21.3% Pearson coefficient outperforms other features. The using of each individual feature cannot yield a satisfactory result. Especially, the removal of the physiological effect based

similarity weaken the prediction performance of model by decreasing the Pearson coefficient by 15.5%. Through various pre-experimental results, we believe that the current combination of medicine-specific features is the one that is much helpful in the semantic similarity calculation of biomedicine.

4.3 Case Study

To study the medicine substitution, we employ MedSim to predict the similarity scores between cefoperazone and other 51 antibiotics. All pairs containing cefoperazone are excluded from training set and considered as test set.

For the antibiotic cefoperazone, Table 4 presents its similar antibiotics whose similarity score is over 0.85. Refer to [23,24], the experiment results show that two antibiotics whose similarity scores over 0.85 can be replaced by each other under normal circumstances. We list the similar antibiotic names, provide the semantic similarity scores between antibiotics and cefoperazone evaluated by MedSim, and present the cases where they can replace each other.

Table 4. Parts of antibiotics similar can replace cefoperazone

Similar antibiotic	Score	Cause where the antibiotic can replace cefoperazone
Cefoxitin	0.865	Respiratory tract infections; Urinary tract infections; Peritonitis; Septicemia; Gynecological infections; Bone, joint, and soft tissue infections
Cefepime	0.864	Respiratory tract infections; Urinary tract infections; Abdominal infections; Reproductive tract infections; Bone, joint, and soft tissue infections
Ceftriaxone	0.860	Lower respiratory tract infection; Urinary tract infections; Complicated intra-abdominal infections; Infections in obstetrics and gynecology; Skin and soft tissue infections; Meningitis
Meropenem	0.851	All infections of cefoperazone, but the has stronger efficacy and wider antibacterial spectrums

Take cefoperazone and ceftriaxone as an example. The indication of cefoperazone is very close to ceftriaxone except disease caused by a few bacteria such as *Pseudomonas aeruginosa*. In the absence of susceptibility testing, doctors can choose either of them to treat most of Gram-negative bacteria infections, such as meningitis, pneumonia and bronchitis. Once the inventory of either is insufficient, our method can help doctors to find a most similar one for replacement.

Another example is cefoperazone and cefpime, both of which have good activity against *Pseudomonas aeruginosa*. However, the combined application of them cannot enhance the efficacy and is considered as drug abuse. Quantifying the similarity of antibiotics, such as listing antibiotics which have similar spectrum of bacterial susceptibility, may help improve public understanding that sometimes

antibiotics combination should be avoided. Thus, medicine semantic similarity measure can ease the increasingly serious problem of antibiotic abuse.

Though meropenem can replace cefoperazone clinically, the semantic similarity score is slightly over 0.85. The reason is that meropenem's indications far exceed cefoperazone. In other words, cefoperazone can not replace meropenem completely. The meropenem will be applied to replace cefoperazone only when the infective bacteria exceeded the antibacterial spectrum of cefoperazone or cefoperazone is ineffective. Otherwise, the replacement of cefoperazone with meropenem is clinically an abuse of antibiotics with higher antibacterial activity.

5 Conclusion

In this study, we propose MedSim, a novel semantic similarity method based on public well-established KGs and large-scale drug corpus. MedSim fully utilizes not only the structural and textual features from the KG but also medicine-specific features. MedSim produces statistically significant improvements over other methods. Examples of case study indicate that calculating the medicine semantic similarity owns a prospect in therapeutic substitution and decreasing the problem of drug abuse. The proposed method is extensible, reproducible and applicable to the KG-based similarity calculation in medical field. Assuming that a drug can be located in both a medical search engine and a bio-medical KG, all the features used in MedSim can be immediately obtained and used to measure other types of medicine in addition to antibiotics. All features used in MedSim can be obtained from the public knowledge source and the labeled dataset is now freely accessible, thus, our method can be conveniently reproduced. In the future, we explore the performance of MedSim in other types of medicine, such as sedative once we get the labeled respective drug dataset.

References

1. Hliaoutakis, A., Varelas, G., Petrakis, E.G.M., Milios, E.: *MedSearch*: a retrieval system for medical information based on semantic similarity. In: Gonzalo, J., Thanos, C., Verdejo, M.F., Carrasco, R.C. (eds.) *ECDL 2006*. LNCS, vol. 4172, pp. 512–515. Springer, Heidelberg (2006). https://doi.org/10.1007/11863878_56
2. Pedersen, T., Pakhomov, S.V., Patwardhan, S., Chute, C.G.: Measures of semantic similarity and relatedness in the biomedical domain. *J. Biomed. Inform.* **40**(3), 288–299 (2007)
3. Resnik, P.: Semantic similarity in a taxonomy: an information-based measure and its application to problems of ambiguity in natural language. *AI Access Found.* **11**, 95–130 (1999)
4. Bollacker, K., Evans, C., Paritosh, P., Sturge, T., Taylor J.: Freebase. In: *Proceedings of SIGMOD* (2008)
5. Wishart, D.S., et al.: DrugBank 5.0: a major update to the DrugBank database for 2018. *Nucleic Acids Res.* **46**(D1), D1074–D1082 (2017)
6. Sun, Y., Han, J., Yan, X., Yu, P.S., Wu, T.: PathSim: meta path-based top-k similarity search in heterogeneous information networks. *Proc. VLDB Endow.* **4**(11), 992–1003 (2011)

7. Shi, C., Li, Y., Yu, P.S., Wu, B.: Constrained-meta-path-based ranking in heterogeneous information network. *Knowl. Inf. Syst.* **49**(2), 719–747 (2016)
8. Resnik, P.: Using information content to evaluate semantic similarity in a taxonomy. In: *International Joint Conference on Artificial Intelligence*, pp. 448–453 (1995)
9. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: *Advances in Neural Information Processing Systems*, vol. 26, pp. 3111–3119 (2013)
10. Zhu, G., Iglesias, C.: Computing semantic similarity of concepts in knowledge graphs. *IEEE Trans. Knowl. Data Eng.* **29**(1), 72–85 (2017)
11. Arup, C., Shrey, S., Pabitra, M., Cyril, S., Nandu, S.S., Muthusamy, C.: SimCat: an entity similarity measure for heterogeneous knowledge graph with categories. In: *Proceedings of the Second ACM IKDD Conference on Data Sciences*, pp. 112–113 (2015)
12. Al-Mubaid, H., Nguyen, H.A.: A cluster-based approach for semantic similarity in the biomedical domain. In: *28th Annual International Conference of the IEEE*, pp. 2713–2717 (2006)
13. Traverso, I., Vidal, M.E., Kämpgen, B., Sure-Vetter, Y.: GADES: a graph-based semantic similarity measure. In: *Proceedings of the 12th International Conference on Semantic Systems*, pp. 101–104. ACM (2016)
14. Hliaoutakis, A.: Semantic similarity measures in MeSH ontology and their application to information retrieval on Medline. Master's thesis (2005)
15. Kuhn, M., Letunic, I., Jensen, L.J., Bork, P.: The SIDER database of drugs and side effects. *Nucleic Acids Res.* **44**(D1), D1075–9 (2015)
16. Pathak, J., Chute, C.G.: Analyzing categorical information in two publicly available drug terminologies: RxNorm and NDF-RT. *J. Am. Med. Inform. Assoc.* **17**(4), 432–439 (2010)
17. Canese, K., Weis, S.: PubMed: the bibliographic database. National Center for Biotechnology Information (2013)
18. Ho, I.W., Lee, C.T., Chen, P.W., Lo, Y.C.: Impact of cumulative antibiograms sub-categorized by origins of infection acquisition on the selection of empirical antimicrobial therapy. *J. Biomed. Lab. Sci.* **27**(1), 10–18 (2015)
19. Hawkyard, C., Koerner, R.: The use of erythromycin as a gastrointestinal prokinetic agent in adult critical care: benefits versus risks authors' response. *J. Antimicrob. Chemother.* **61**(1), 227–228 (2007)
20. Bryan, P., Rami A.R., Steven, S.: DeepWalk. In: *Proceedings of SIGKDD* (2014)
21. Gabrilovich, E., Markovitch, S.: Computing semantic relatedness using Wikipedia-based explicit semantic analysis. In: *International Joint Conference on Artificial Intelligence*, pp. 1606–1611 (2007)
22. Robertson, S.: The probabilistic relevance framework: BM25 and beyond. *Found. Trends Inf. Retr.* **3**(4), 333–389 (2010)
23. Ho, P., Wong, S.: Reducing bacterial resistance with IMPACT-Interhospital Multi-disciplinary Programme on Antimicrobial ChemoTherapy, 4th edn. Meteoritics And Planetarience, pp. 1–176 (2012)
24. Antibiotic guidelines. http://www.hopkinsmedicine.org/amp/guidelines/antibiotic_guidelines.pdf. Accessed 25 Jan 2018



A Sequence Transformation Model for Chinese Named Entity Recognition

Qingyue Wang^{1,3}, Yanjing Song², Hao Liu², Yanan Cao³(✉),
Yanbing Liu³, and Li Guo³

¹ School of Cyber Security, University of Chinese Academy of Sciences,
Beijing, China

qingyue.wang2018@gmail.com

² Software Institute, Beijing Institute of Technology, Beijing, China
yanjing.song2018@gmail.com, huakaicary@gmail.com

³ Institute of Information Engineering, Chinese Academy of Sciences,
Beijing, China

{caoyanan, liuyanbing, guoli}@iie.ac.cn

Abstract. Chinese Named Entity Recognition (NER), as one of basic natural language processing tasks, is still a tough problem due to Chinese polysemy and complexity. In recent years, most of previous works regard NER as a sequence tagging task, including statistical models and deep learning methods. In this paper, we innovatively consider NER as a sequence transformation task in which the unlabeled sequences (source texts) are converted to labeled sequences (NER labels). In order to model this sequence transformation task, we design a sequence-to-sequence neural network, which combines a Conditional Random Fields (CRF) layer to efficiently use sentence level tag information and the attention mechanism to capture the most important semantic information of the encoded sequence. In experiments, we evaluate different models both on a standard corpus consisting of news data and an unnormalized one consisting of short messages. Experimental results showed that our model outperforms the state-of-the-art methods on recognizing short interdependence entity.

Keywords: Named Entity Recognition · Deep learning
Sequence to sequence neural network · Conditional Random Fields

1 Introduction

Named Entity Recognition (NER) is used to accurately identify a series of entities from text, such as person, location and organization, which can be used for senior natural language processing (NLP) applications.

Most related works regard NER as a sequence tagging task. Typical statistical models include Hidden Markov Model (HMM) [13], Conditional Random Fields (CRF) [17] and etc. They still suffer from extracting effective grammatical features and templates manually. As more and more systems using neural models have achieved good performances in different NLP tasks, deep neural network on sequence tasks raises continuing concern. Collobert [6] firstly addressed the sequence tagging

problems in an end-to-end way, which tried to pre-process features as little as possible and designed a multilayer neural network architecture for Word Segmentation, Chunking and Named Entity Recognition. However, its performance is limited by the fixed size window of words although the neural language is closely related to the context. Most recently, Huang [20] combined a bidirectional Long Short-Term Memory [11] (LSTM) network and a CRF layer, called BiLSTM-CRF, which produced state-of-art accuracy on several NLP tagging tasks. In this model, BiLSTM uses both past and future information of input and CRF layer utilizes sentence-level tags. Ma [8] introduced an end-to-end network architecture which combines bidirectional LSTM, convolutional neural networks (CNN) and CRF, and it benefits from both word-level and character-level representations.

Unlike previous works, we regard NER as a sequence transformation task in this paper. In order to properly model this task, we propose a variety of sequence-to-sequence (seq2seq) neural network models. In the baseline seq2seq model, we use a BiLSTM encoder and a LSTM decoder to capture the context information for LSTM's good ability to solve long-term dependencies in source text. In the upgraded model, we combined the seq2seq model with a CRF layer, which can utilize the past and future tags to predict the current tag with high precision. Besides, we utilize an attention mechanism to both above models, which is conditioned on a distinct context vector for each target label, making the decoder pay more attention on current context information during predicting sequence. These models are all evaluated on a standard corpus (People's Daily news) and a short message corpus we constructed. And the experiments results showed that our model reaches a good performance especially on short dependence entity.

Our contributions can be summarized as follows. (1) We design several sequence-to-sequence models for Chinese Named Entity Recognition. As far as we know, we are among the first endeavors to resolve the NER problem in this way. (2) We explore the effectiveness of attention mechanism on our models for Chinese Named Entity Recognition. (3) We systematically compare the performance of existing models on both short messages and news texts for NER. And we show that our model based seq2seq-CRF-attention can produce state-of-the-art (or close to) F1 scores on recognizing person, location and organization.

The remainder of this paper is organized as follows. Section 2 discusses related work. Section 3 describes the model we propose in detail. Section 4 introduces the experiments and analysis results on different methods and entities. Section 5 draws conclusions finally.

2 Related Work

The earliest Named Entity Recognition method was rule-based recognition, which relied on the language experts or domain experts to specify effective grammar rules such as gazetteers, costing a lot of time and energy. For sequence labeling task, Hidden Markov Model [13] (HMM), Support Vector Machine [16] (SVM), maximum entropy

Markov models [15] (MEMMS) and Conditional Random Fields [17, 19] (CRF) once achieved good results on NER. In recent years, several neural architectures start showing great learning power. Deep Neural Network proposed by Collobert [6] introduced a radically approach trying to preprocess features as little as possible and used a multilayer neural network-based windows and sentences. Lample [7] presented a LSTM-CRF architect with a char-LSTM layer learning spelling features from supervised corpus and didn't use any additional resources. Following the idea, Dong [9] was the first to investigate Chinese radical-level representation in BiLSTM-CRF architecture and got better performance without carefully designed features. Ma [8] proposed a BiLSTM-CNNs-CRF architecture using CNNs to model character-level information. To apply neural network to natural language, word embedding [4, 10] is used to convert language tokens to vectors, which greatly help express word meaning in space and improve the performance of many NLP applications.

In machine translation, Sutskever [1] proposed an encoder and a decoder for each language. This model is jointly trained to maximize the probability of a correct translation. Hermann [3] used the similar neural encoder-decoder model in question answering and Nallapati [5] proposed a neural network model in abstractive text summarization using sequence-to-sequence. Bahdanau [2] achieved a novel neural network model based on attentional encoder-decoder model for machine translation. Inspired by this mechanism, Paulus [12] introduced a neural network model with an intra-attention and a new training method that combines standard supervised word prediction in abstractive summarization. Unlike in machine translation and speech recognition, alignment is explicit in some NLP applications. Liu and Lane [18] described their approach introducing attention to the alignment-based RNN models for joint intent detection and slot filling. Our model basically follows their idea, but we modify the model to solve NER problem.

3 Proposed Methods

We combine the seq2seq model, attention mechanism and a CRF network to form a seq2seq-Attention-CRF model, which is illustrated in Fig. 1. Given an input sentence denoted by $x = \{x_1, x_2, \dots, x_T\}$, x_t represents the t -th character (or word) at time step t . We use a bidirectional LSTM network (BiLSTM) to encode the input sentence, and a LSTM network to decode the hidden state h_t and vector c_t from the encoder. The vector c_t computed by attention mechanism is used to capture the information of the encoded sequence. After decoding, CRF layer utilizes the probability generated from the decoder and transition matrix to predict optimal tagging sequence. The output $y = \{y_1, y_2, \dots, y_T\}$ represents labeled sequence corresponding to the input. Here, we use the most common tagging scheme named IOB (Inside, Beginning, Outside). Next, we introduce the components of our model respectively: aligned encoder-decoder, encoder-decoder based attention and CRF layer.

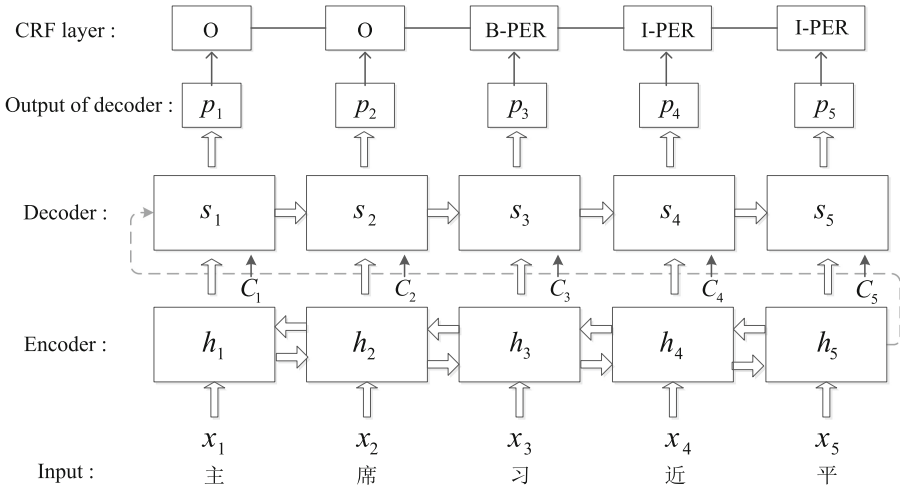


Fig. 1. Our model for Chinese Named Recognition based character-level. We use a bidirectional LSTM as an encoder, a unidirectional LSTM as a decoder, attention mechanism in seq2seq model and a CRF layer after decoding.

3.1 Aligned Encoder-Decoder

Here, we briefly describe the underlying framework, called encoder-decoder which learns to align and recognize name entity. The encoder and decoder are two separate RNNs.

On encoder side, the model reads the input sequence with a bidirectional LSTM encoder. For a given sentence x containing T characters, each character is represented as a d -dimensional vector. A LSTM computes a representation \vec{h}_t of the left context of the sentence at each time t . Similarly, the right context \vec{h}_t starting from the end of the sentence provides the future information of input. The final encoder hidden state h_t at each time step t is obtained by concatenating its left and right context representations $h_t = [\vec{h}_t, \vec{h}_t]$.

On decoder side, we use a unidirectional LSTM because the encoder with forward and backward LSTM has carried entire information of the sequence. We initialize the decoder hidden state with $s_0 = h_T$. At each decoding step t , the decoder hidden state s_t is equal to a function of the previous emitted label y_{t-1} , the aligned encoder hidden state h_t , and the context vector c_t :

$$s_t = f(s_{t-1}, y_{t-1}, h_t, c_t) \tag{1}$$

f is a nonlinear function and we use LSTM as f . The context vector c_t computed by attention mechanism will be introduced in next section.

3.2 Encoder-Decoder Based Attention

By allowing a model to automatically search for parts of a source sentence that are relevant to predicting a target word, attention mechanism can be spread throughout the sequence of annotations and retrieved by the decoder accordingly. Attention mechanism has shown promising results in many other NLP tasks such as machine translation, speech recognition and etc. Inspired by these works, we introduce the attention mechanism in neural machine translation which is proposed by Bahdanau [2]. The illustration of the attention mechanism in our model is shown in Fig. 2.

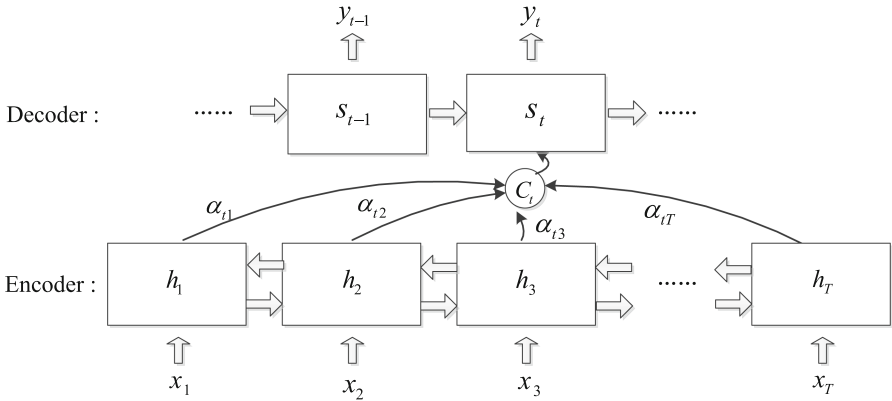


Fig. 2. The graphical illustration of the attention mechanism trying to generate the t -th predicted label given a source sentence x .

At each decoding step t , an attention function is used to attend over specific part of the encoded input sequence. The context vector c_t input to the decoder is computed as a weighted sum of the encoder hidden state h_j :

$$c_t = \sum_{j=1}^{T_x} \alpha_{tj} h_j \quad (2)$$

The weight α_{tj} of each hidden state h_j is computed by:

$$\alpha_{tj} = \frac{\exp(e_{tj})}{\sum_{k=1}^{T_x} \exp(e_{tk})} \quad (3)$$

$$e_{tj} = a(s_{t-1}, h_j) \quad (4)$$

where a is a feedforward neural network which is jointly trained with all the other components of the proposed model.

3.3 CRF Layer

It has been shown that CRF can produce higher tagging accuracy for part-of-speech (POS), chunking and NER, because it can efficiently use sentence level tag information. A CRF layer has a state transition matrix that can be trained with other parameters in the seq2seq-CRF-attention network.

We consider that the probability matrix $f_{\theta}([x]_1^T)$ is output by the network we proposed. The element $[f_{\theta}]_{[i],t}$ of the matrix is the score output by the network with parameters θ , for sentence $[x]_1^T$ and for the i -th tag at the t -th word. In our model, we note the new parameters as $\tilde{\theta} = \theta \cup \{[A]_{i,j} \forall i, j\}$. The transition probability matrix $[A]_{i,j}$ represents the transition from the i -th tag to j -th tag. The final score of a sentence $[x]_1^T$ is defined as follows:

$$s([x]_1^T, [i]_1^T, \tilde{\theta}) = \sum_{t=1}^T ([A]_{[i]_{t-1}, [i]_t} + [f_{\theta}]_{[i]_t, t}) \quad (5)$$

To choose the optimal labeling sequence, we make use of the equation by the principle of maximum likelihood estimation. The best sequence can be computed as follows:

$$[\hat{i}]_1^T = \arg \max(s([x]_1^T, [i]_1^T, \tilde{\theta})) \quad (6)$$

4 Experiments and Result

4.1 Datasets and Evaluation

We evaluate the proposed approach on both unnormalized text (short messages) and standard text (People's Daily News). It should be noted that the short messages dataset contains more noise data including inform nicknames and wrong characters compared with news corpus. We estimate the system performance using precision (P), recall (R), F1 scores (F1) and IOB tagging scheme.

Short Messages. This corpus includes 200,000 messages, and the average text length is about 60 characters. We use 160,000 messages as a training dataset and 40,000 messages as a testing one. There are a few organization entities in this corpus, so we mainly recognized two types of entities: person and location.

People's Daily News. This dataset contains the whole 2000 year's news, and we regard the first ten months including 431289 sentences as a training dataset and the rest 98579 sentences as a testing one. In this corpus, we recognized three types of entities: person, location and organization.

4.2 Comparative Methods

We aim to evaluate the effectiveness of our proposed seq2seq models. In experiments, we use several typical statistical machine learning methods and neural network models

as comparative methods, including state-of-the-art models. Here is a brief induction to these methods.

HMM. Hidden Markov model [13] is a statistical Markov model in which the system being modeled is assumed to be a Markov process with unobserved (i.e. hidden) states. The sequence of tokens generated by an HMM gives some information about the sequence of states so that it is especially known for their application in sequence problems such as speech and part-of-speech tagging.

CRF. Conditional Random Fields applied for labeling sequential data is a probabilistic model, and it also is the best statistical model on NER. In practice, we use the CRF++ package¹, a customizable and open source implementation of CRF, as an indispensable part of comparative methods.

BiLSTM+CRF. This method combining BiLSTM with CRF layer gains advantages of both neural network model and probabilistic model, and it also achieves state-of-the-art performance on tagging problems. We apply the same network architecture described by Huang [20] on Chinese Named Entity Recognition.

Seq2seq. This method is one of our baseline models. Since seq2seq was proposed, it has been applied in many sequence problems such as machine translation and image captioning. As a baseline model, it is designed as bidirectional LSTM encoder and unidirectional LSTM decoder.

Seq2seq+Attention. It is another model of our baselines. In experiments, we extend seq2seq model (mentioned above) with attention following the design of Bahdanau [2]. We compare this model with seq2seq to verify the effectiveness of attention on NER problem.

4.3 Implementation Details

The implementation of two LSTM follows the design in [22]. We set the number of units in LSTM cell as 200. Dropout rate 0.5 is applied to the non-recurrent connections [22] during model training for regularization. We adopt Adam optimization algorithm [21] starting with an empirical learning-rate of 0.001. Word embedding of size 256 are randomly initialized and fine-tuned during mini-batch training with batch size of 16. To avoid generating an oversized vocabulary, we delete the low-frequency character (or word) which appears less than 5 times in the corpus. The maximum norm for gradient clipping is set to 5. Our implementation is fully based on tensorflow1.4 [23]. A script tool² called “colleval.pl” is used to evaluate the performance of NER.

4.4 Results and Analysis

We use our model and all comparative methods to recognize various entities on two datasets. Tables 1 and 2 respectively show the results of recognizing person and

¹ <https://www.findbestopensource.com/product/crfpp>.

² <http://www.cnts.ua.ac.be/conll2000/chunking/>.

location on the Short Message corpus, while Table 3, 4 and 5 respectively show the results of recognizing person, location and organization on the People’s Daily News. We analyze the results in the following.

Evaluation on Input Representation. In this paper, we both use the character-level and word-level input for all models. From Table 1, we can find that all neural network models using character-level representation get higher F1 scores than those using word-level one, while systems using word-level achieve better results on location from Table 2. It maybe because that short messages, which tends to be spoken language and informal expressions, generally contains more noise such as nicknames and spelling mistakes. Chinese person entity is a loose internal structure that words or characters are independent of each other. Character-level avoids the interference of word meanings. Location, differing with person, has strong meanings and consists of two or more words. Although we design the encoder using BiLSTM to connect the context, the system with character-level still can’t understand the interdependence and meanings of words. We can also observe that the systems using word-level perform better on both person and location for News (Tables 3 and 4). News is a kind of written language with formal expressions, which means it contains less wrong word segmentation than short messages. In other words, only accurate word segmentation is helpful to recognize entity. Similar to location, organization is usually composed of several practical significance words. However, this dependence between words becomes weaker and weaker as the length of organization increasing. This also explains why the systems with character-level outperform again on organization (Table 5).

Table 1. Proposed approaches on Short Messages for person.

Model	Character-level			Word-level		
	P	R	F1	P	R	F1
HMM	78.24	87.47	82.60	85.66	75.75	80.04
CRF	85.09	90.48	87.70	86.43	80.52	83.37
Seq2seq	92.32	92.32	92.32	89.43	88.18	88.80
Seq2seq+Attention	92.19	93.54	92.86	90.72	89.28	90.00
BiLSTM+CRF	93.99	94.60	94.30*	91.91	90.58	91.24*
Seq2seq+CRF	93.52	92.66	93.09	90.25	89.02	89.63
Seq2seq+Attention+CRF	92.74	94.90	93.80	92.75	89.62	91.16

Table 2. Proposed approaches on Short Messages for location.

Model	Character-level			Word-level		
	P	R	F1	P	R	F1
HMM	76.20	83.22	79.55	93.73	92.08	92.90
CRF	91.52	94.17	92.82	95.51	91.06	93.23
Seq2seq	93.54	93.12	93.33	95.95	95.89	95.92
Seq2seq+Attention	95.42	96.49	95.95	96.78	96.76	96.77
BiLSTM+CRF	96.29	96.52	96.41	97.18	96.98	97.08
Seq2seq+CRF	94.43	93.93	94.18	97.16	95.64	96.39
Seq2seq+Attention+CRF	96.41	96.55	96.48*	97.00	97.24	97.12*

Effectiveness of the CRF Layer. We find that CRF outperforms HMM both on character-level and word-level for recognizing all entities (From Tables 1, 2, 3, 4 and 5). CRF doesn't have the strict independence assumption of HMM so it can accommodate much context information, without its flexible features. Comparing Seq2seq+Attention model with Seq2seq+Attention+CRF model, we find that the performance with CRF is improved evidently. For example, the system (character-level) reached 94.19% F1 with CRF but only 92.70% F1 without CRF on location (Table 4). The reason is that CRF using both past and future labels avoids generating wrong tagging sequence to a great degree. Besides, it is easy to see that the improvement is more evident on character-level representation, because the models need to generate much more tags for input characters than words. So the systems with CRF using character-level can work better on sequence tagging problems.

Table 3. Proposed approaches on People's Daily News for person.

Model	Character-level			Word-level		
	P	R	F1	P	R	F1
HMM	68.45	69.76	69.10	94.35	88.81	91.49
CRF	89.43	89.84	89.63	96.10	86.71	91.16
Seq2seq	88.79	84.93	86.82	94.73	90.28	92.45
Seq2seq+Attention	95.35	94.60	94.97	96.70	96.07	96.38
BiLSTM+CRF	97.02	94.98	95.99*	97.75	95.67	96.70*
Seq2seq+CRF	92.46	88.98	90.68	95.23	92.05	93.61
Seq2seq+Attention+CRF	96.57	94.64	95.59	96.58	96.14	96.36

Table 4. Proposed approaches on People's Daily News for location.

Model	Character-level			Word-level		
	P	R	F1	P	R	F1
HMM	58.31	66.69	62.22	87.34	89.91	88.61
CRF	77.43	84.34	80.73	92.15	80.40	85.87
Seq2seq	82.24	77.76	79.94	92.56	89.20	90.85
Seq2seq+Attention	92.58	92.82	92.70	95.62	93.28	94.43
BiLSTM+CRF	93.41	93.32	93.36	95.82	93.48	94.64
Seq2seq+CRF	88.06	84.01	85.99	92.88	90.73	91.79
Seq2seq+Attention+CRF	94.29	94.09	94.19*	94.95	94.68	94.81*

Evaluation on Sequence Transformation Models. For both People's Daily News (Tables 3, 4 and 5) and Short Messages (Tables 1 and 2), the Seq2seq+Attention+CRF model outperform BiLSTM+CRF on location, but ranks only second to BiLSTM+CRF on person and organization. As we mentioned above, person entity generally contains less inside information while location contains much internal dependence between atomic words. Organization is usually longer than location, which may increase the accumulation of errors in decoding. According to above analysis, we conclude that the

Seq2seq with attention and CRF model is very good at recognizing short strong inter-dependent entity such as location. Even to other entities, such as person and organization, the performance of our model is much close to the best results (BiLSTM+CRF).

Table 5. Proposed approaches on People’s Daily News for organization.

Model	Character-level			Word-level		
	P	R	F1	P	R	F1
HMM	47.99	50.86	49.38	77.56	81.86	79.65
CRF	50.22	71.43	58.98	80.42	84.11	82.22
Seq2seq	83.39	83.39	83.39	92.42	92.98	92.70
Seq2seq+Attention	91.79	93.68	92.72	92.47	93.16	92.81
BiLSTM+CRF	95.22	94.57	94.89*	94.52	93.84	94.18*
Seq2seq+CRF	91.79	87.01	89.34	92.82	90.09	91.44
Seq2seq+Attention+CRF	95.21	93.83	94.52	94.13	95.07	94.09

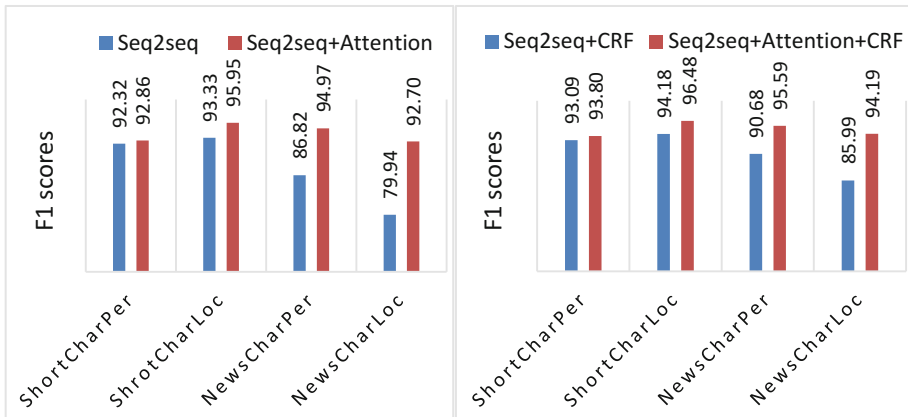


Fig. 3. The graphical illustrations of the attention mechanism using different corpus on F1 scores.

Effectiveness of the Attention Mechanism. To estimate the effectiveness of the attention mechanism, we compare the results using Seq2seq model and Seq2seq+Attention model, Seq2seq+CRF model and Seq2seq+Attention+CRF model on different corpus for person and location. The contrastive results on F1 score are shown in Fig. 3. By adding attention mechanism, the systems do really prompt the performance of NER. Besides, it is easily to see that the promotion is more obvious on News corpus. As we mentioned above, short messages includes more noise data because of its informal expression. Models using attention on informal text may gain wrong surrounding information during recognition compared with formal text sometimes. So we conclude that seq2seq can’t capture enough contextual information very well which can be compensated by the attention, especially on formal text.

5 Conclusion

In this paper, we explored a sequence transformation framework for Chinese Named Entity. We also systematically compared the performance of different NER systems, showing that our model achieves a good performance especially on short interdependence entity.

There are still some problems need to be considered. Firstly, word segmentation information is helpful for formal text in NER but not for informal, which can be considered how to join word-level and character-level embedding in the further NER research. Secondly, our model is a supervised method relying on a large number corpus that is not suitable to small-labeling-data such as social media. So, it is necessary to study a semi-supervised framework or transfer learning framework for NER.

Acknowledgement. This work was supported by the National Key Research and Development program of China (No. 2016YFB0801300), the National Natural Science Foundation of China grants (No. 61602466).

References

1. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. **4**, 3104–3112 (2014)
2. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. *Computer Science* (2014)
3. Hermann, K.M., Kočiský, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., et al.: Teaching machines to read and comprehend, pp. 1693–1701 (2015)
4. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J.: Distributed representations of words and phrases and their compositionality. **26**, 3111–3119 (2013)
5. Nallapati, R., Zhou, B., Santos, C.N.D., Gulcehre, C., Xiang, B.: Abstractive text summarization using sequence-to-sequence RNNs and beyond (2016)
6. Collbert, R., Weston, J., Bottou, L.: Natural language processing (almost) from scratch. *J. Mach. Learn. Res.* **12**, 2493–2537 (2011)
7. Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., Dyer, C.: Neural architectures for named entity recognition, pp. 260–270 (2016)
8. Ma, X., Hovy, E.: End-to-end sequence labeling via bi-directional LSTM-CNNs-CRF. [arXiv:1603.01354v4](https://arxiv.org/abs/1603.01354v4) (2016)
9. Dong, C., Zhang, J., Zong, C., Hattori, M., Di, H.: Character-based LSTM-CRF with radical-level features for Chinese named entity recognition. In: Lin, C.-Y., Xue, N., Zhao, D., Huang, X., Feng, Y. (eds.) ICCPOL/NLPCC 2016. LNCS (LNAI), vol. 10102, pp. 239–250. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-50496-4_20
10. Pennington, J., Socher, R., Manning, C.: Glove: global vectors for word representation. In: Conference on Empirical Methods in Natural Language Processing, pp. 1532–1543 (2014)
11. Hochreiter, S., Schmidhuber, J.: Long short-term memory. *Neural Comput.* **9**(8), 1735–1780 (1997)
12. Paulus, R., Xiong, C., Socher, R.: A deep reinforced model for abstractive summarization. *arXiv preprint arXiv:1705.04304v3* (2017)

13. Su, J., Su, J.: Named entity recognition using an HMM-based chunk tagger. In: Meeting on Association for Computational Linguistics, pp. 473–480. Association for Computational Linguistics (2002)
14. Borthwick, A.: A Maximum Entropy Approach to Named Entity Recognition. New York University (1999)
15. Hai, L.C., Ng, H.T.: Named entity recognition: a maximum entropy approach using global information. In: International Conference on Computational Linguistics, pp. 1–7. Association for Computational Linguistics (2002)
16. Li, L., Mao, T., Huang, D., Yang, Y.: Hybrid models for Chinese named entity recognition. In: Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing, pp. 72–78 (2006)
17. Chen, A., Peng, F., Shan, R., Sun, G.: Chinese named entity recognition with conditional probabilistic models. In: Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing, pp. 173–176 (2006)
18. Liu, B., Lane, I.: Attention-based recurrent neural network models for joint intent detection and slot filling (2016)
19. Lafferty, J.D., McCallum, A., Pereira, F.C.N.: Conditional random fields: probabilistic models for segmenting and labeling sequence data. In: Eighteenth International Conference on Machine Learning, vol. 3, pp. 282–289. Morgan Kaufmann Publishers Inc. (2001)
20. Huang, Z., Xu, W., Yu, K.: Bidirectional LSTM-CRF models for sequence tagging. Computer Science (2015)
21. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. Computer Science. arXiv preprint [arXiv:1412.6980](https://arxiv.org/abs/1412.6980) (2014)
22. Zaremba, W., Sutskever, I., Vinyals, O.: Recurrent neural network regularization (2014)
23. Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C.: TensorFlow: large-scale machine learning on heterogeneous distributed systems (2016)



An Incremental Reasoning Algorithm for Large Scale Knowledge Graph

Yifei Wang and Jie Luo^(✉)

State Key Laboratory of Software Development Environment,
School of Computer Science and Engineering, Beihang University,
Beijing 100191, China

{wangyifei,luojie}@nlsde.buaa.edu.cn

Abstract. Knowledge graphs usually contain much implicit semantic information, which needs to be further mined through semantic inference. Current algorithms can effectively accomplish such task, however they often require a full re-reasoning even when only a few new triples is added to expand the knowledge graph. In this paper, we propose an incremental reasoning algorithm which can effectively avoid re-reasoning over the entire knowledge graph while keeping the relative completeness of the final deduction results. Key to our approach is the filter algorithms which reduce the scale of data that need to be considered and a delay strategy which limit the number of time-consuming iterations while still preserve relative completeness. Extensive experiments and comprehensive evaluations are conducted and experimental results prove that our methods significantly outperform re-reasoning methods.

Keywords: Knowledge reasoning · Incremental reasoning
Knowledge graph · OWL2 RL

1 Introduction

In the past decades, knowledge graph has been widely used in various fields, such as question answering and search engine, due to the fact that knowledge graph contains rich semantic information. However, there is still much implicit semantic information, which can not be directly used and needs to be further mined. Both probabilistic model-based algorithms, like Probabilistic Soft Logic [9], and rule-based algorithms can be applied to such task. In face of large scale knowledge graph, rule-based algorithms have achieved significant performance and thus become mainstream methods.

Rule-based algorithms rely more on a fixed set of logic rules. Early rule-based reasoning systems, such as Jena, Sesame, Pellet [4, 11, 13], are mostly running on a single machine. However, as the size of knowledge graph grows, many distributed rule-based algorithms appear. Urbani et al. [16] propose a reasoning algorithm based on RDFS using MapReduce computational models [5]. Later, they extend the system to a new OWL Horst semantic reasoning system

called WebPIE [15]. Rong Gu et al. design Cichlid reasoning system [6] based on RDFS/OWL using Spark computing platform [18]. Kim and Park [8] also design a distributed reasoning algorithm based on Spark [18] with OWL Horst rules. Besides, there are reasoning algorithms designed for streaming data [1–3,14]. To make it easier for semantic reasoning researchers to exchange ideas, Bijan Parsia et al. set up the OWL Reasoner Evaluation [12], which is a competition for common reasoning tasks. These algorithms work well for general large scale knowledge graph reasoning tasks, however, for cases where the knowledge graph is on continuous expansion with new triples, they need to perform a full reasoning after each expansion of the knowledge graph which consumes a lot of time and causes a lot of redundant computation.

Therefore, this paper proposes an incremental reasoning algorithm on the basis of KGRL [17], a large-scale knowledge graph reasoning algorithm based on OWL2 RL. We first introduced an irrelevant triples filtering process to reduce the size of data that needs to be considered. Then we introduce a delay strategy which preserves relative completeness of the incremental reasoning results to improve the performance. Experiments on the DBpedia [10] and LUBM [7] datasets show that, when incrementally adding a relatively small number of new triples to the original knowledge graph, our algorithm is much faster than a full re-reasoning using KGRL [17] and the loss caused by the delay strategy is relatively small.

2 Previous Work

In our previous work, we propose a rule-based reasoning algorithm for OWL2 RL, named KGRL, which is targeted for performing reasoning over large scale knowledge graphs. OWL2 RL made a good trade-off between efficiency and expressiveness and is a proper choice for scalable reasoning tasks without lose too much expressive power. Because of the complexity of rules in OWL2 RL, iterative application of rules until no new triples is produced is necessary to get the closure of the input knowledge graph. After analyzing the relationships between rules of OWL2 RL, we divided these rules into 5 groups to minimize the number of application iterations of rules. Figure 1 shows the 5 groups and their applying

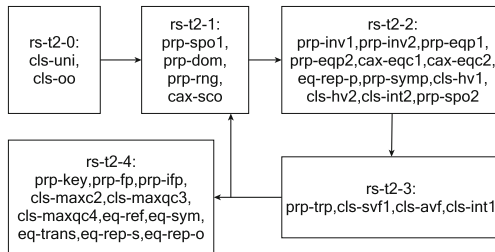


Fig. 1. Interdependency of groups of OWL2 RL rules

orders. Experiments show that KGRL gain great advantages both in efficiency and the number of deduced triples.

Although the rule-based reasoning algorithm KGRL is proved to be efficient for reasoning over large scale knowledge graphs, there are still some problems need to be solved when we consider an incremental reasoning algorithm.

3 Incremental Reasoning Algorithm

The KGRL algorithm is aimed for deducing the theory closure of a given knowledge graph under OWL2 RL rule set. But when the given knowledge graph is updated, e.g. new triples are added to it, how can we deduce the theory closure of the updated knowledge graph? The simplest answer is that update the knowledge graph and then perform a full reasoning by KGRL again on the updated knowledge graph. However, for large scale knowledge graphs (for instance, knowledge graphs which contain trillions of triples), a full reasoning consumes too much time and perform a full reasoning for each small expansion to the knowledge graph is not acceptable. Thus, incremental reasoning algorithm is required to deduce the theory closure $Th(K_1)$ of the updated knowledge graph $K_1 = K_0 \cup T_1$ from the theory closure $Th(K_0)$ of the original knowledge graph K_0 and the addition triples T_1 for expanding K_0 , as described in Fig. 2.

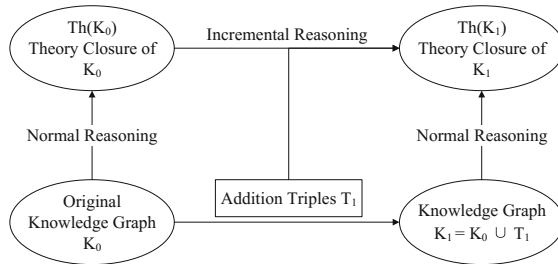


Fig. 2. The problem of incremental reasoning for theory closures

In this section, we shall explain the challenges for design an incremental reasoning algorithm for large scale knowledge graph, our strategies for handling them, and details of the proposed incremental reasoning algorithm.

For convenience, we shall treat a knowledge graph as a set of triples, denote knowledge graphs as K_i , theory closure of K_i as $Th(K_i)$, the incremental additions to knowledge graph as T_i , and new triples deduced by incremental reasoning as N_i .

For large scale knowledge graph, the size $|K_0|$ of the initial knowledge graph K_0 is far larger than the size $|T_i|$ of each incremental expansion (addition) T_i to K_{i-1} , where $i \geq 1$.

3.1 Challenges

There are two key challenges for designing a successful incremental reasoning algorithm. One is to make the incremental reasoning algorithm efficiently deducing new triples based the results of the previous incremental reasoning and the addition triples for incremental expansion of the knowledge graph. The key issue is to avoid iterate through the whole knowledge graph for rule applying. The other is to preserve the completeness of the reasoning results, i.e. make sure the results of the incremental reasoning will eventually convergent to the results of KGRL reasoning, which is the theory closure of the knowledge graph after all incremental expansion under OWL2 RL rule set.

3.2 Approaches

In order to deal with the above two challenges, we put forward two approaches for improving the performance of incremental reasoning and preserving the completeness of the results, which are further described in the following.

Irrelevant Triples Filtering. Based on the number of triples in the premise, the OWL2 RL rule set \mathcal{R} can be divided into two sets. The first set consists of rules whose premises contain single triple, denoted as \mathcal{R}_S . For rules in this set, they are only need to be applied to new triples which have never been applied before. For example, consider the addition T_1 to the knowledge graph K_0 , rules in \mathcal{R}_S only need to apply to triples in T_1 , since all triples can be deduced by these rules from K_0 are already contained in $\text{Th}(K_0)$.

The second set consists of rules whose premises contain multiple triples, denoted as \mathcal{R}_M . Applications of these rules are the tricky part for efficient incremental reasoning. There are eight rules in \mathcal{R}_M as listed in Table 1. For these rules, the triples in the premises can be from both the previous closure $\text{Th}(K_{i-1})$ and the incremental addition T_i to K_{i-1} , and at least one of the triples must from T_i . Otherwise, all triples in the premise are from $\text{Th}(K_{i-1})$. Therefore the resulting triples of applying this rule are already contained in $\text{Th}(K_{i-1})$ by the definition of theory closure and it is not necessary to perform the same reasoning again.

Because the size of T_i is relatively small, it is easy and efficient to find triples in it, which match the pattern of the premise of a rule in \mathcal{R}_M . But for the triples in $\text{Th}(K_{i-1})$, it is too large to iterate through to try all possible combinations. However, since the rules only need to apply to premises which contain at least one triple from T_i , we can use this restriction and the small amount of triples in T_i to filter out large part of irrelevant triples in $\text{Th}(K_{i-1})$ and improve the efficiency of reasoning with rules in \mathcal{R}_M .

These two sets of rules can be treated separately in each round of rule application, but the deduced new triples (triples which do not occurred previously) must be combined together and treated as a new addition to the knowledge graph for next round of rule application. This is because of the mutual interdependence between rules of OWL2 RL, which requires iteratively applying all rules until no new triple can be deduced.

Table 1. OWL RL2 rules whose premises contain multiple triples

Rule	If	Then
prp-fp	$T(?p, \text{rdf:type}, \text{owl:FunctionalProperty})$ $T(?x, ?p, ?y_1), T(?x, ?p, ?y_2)$	$T(?y_1, \text{owl:sameAs}, ?y_2)$
prp-ifp	$T(?p, \text{rdf:type}, \text{owl:InverseFunctionalProperty})$ $T(?x_1, ?p, ?y), T(?x_2, ?p, ?y)$	$T(?x_1, \text{owl:sameAs}, ?x_2)$
prp-trp	$T(?p, \text{rdf:type}, \text{owl:TransitiveProperty})$ $T(?x, ?p, ?y), T(?y, ?p, ?z)$	$T(?x, ?p, ?z)$
cls-maxc2	$T(?x, \text{owl:maxCardinality}, "1" \wedge \text{xsd:nonNegativeInteger})$ $T(?x, \text{owl:onProperty}, ?p), T(?u, \text{rdf:type}, ?x)$ $T(?u, ?p, ?y_1), T(?u, ?p, ?y_2)$	$T(?y_1, \text{owl:sameAs}, ?y_2)$
prp-spo2	$T(?p, \text{owl:propertyChainAxiom}, ?x), \text{List}[?x, ?p_1, \dots, ?p_n]$ $T(?u_1, ?p_1, ?u_2), \dots, T(?u_n, ?p_n, ?u_{n+1})$	$T(?u_1, ?p, ?u_{n+1})$
cls-maxqc3	$T(?x, \text{owl:maxQualifiedCardinality}, "1" \wedge \text{xsd:nonNegativeInteger})$ $T(?x, \text{owl:onProperty}, ?p), T(?x, \text{owl:onClass}, ?c)$	$T(?y_1, \text{owl:sameAs}, ?y_2)$
cls-maxqc4	$T(?x, \text{owl:maxQualifiedCardinality}, "1" \wedge \text{xsd:nonNegativeInteger})$ $T(?x, \text{owl:onProperty}, ?p), T(?x, \text{owl:onClass}, \text{owl:Thing})$ $T(?u, \text{rdf:type}, ?x), T(?u, ?p, ?y_1), T(?u, ?p, ?y_2)$	$T(?y_1, \text{owl:sameAs}, ?y_2)$
prp-key	$T(?c, \text{owl:hasKey}, ?u), \text{List}[?u, ?p_1, \dots, ?p_n]$ $T(?x, \text{rdf:type}, ?c), T(?x, ?p_1, ?z_1), \dots, T(?x, ?p_n, ?z_n)$ $T(?y, \text{rdf:type}, ?c), T(?y, ?p_1, ?z_1), \dots, T(?y, ?p_n, ?z_n)$	$T(?x, \text{owl:sameAs}, ?y)$

Relative Completeness Preserving. The second challenge for designing incremental reasoning algorithm is to make sure that the reasoning results convergent to the closure of the knowledge graph which consists of the original knowledge graph and all addition triples. One simple solution is to preserve the completeness for each incremental reasoning, i.e. iteratively apply the above two sets of rules until no new triple is deduced. However, we found this approach quite deficiency for distributed reasoning of large scale knowledge graph. The reason is that, for distributed implementation of rule based reasoning, the runtime of each iteration of rule application consists of a constant base runtime which is determined by the distributed computation platform (e.g. the Spark clusters for this paper) and the runtime of rule applying. So the total base runtime cost by the platform and the number of iterations have a linear positive correlation. Thus, as the number of required iterations for rule applying increase, the efficiency of the rule based reasoning algorithm decrease. The number of iterations of rule applying should be minimized in order to improve the efficiency.

It can also be observed that, as the number of iteration increase, the number of new triples deduced in this iteration decrease rapidly. That is, most of the new triples can be deduced in the first few rounds of rule application, the rest rounds of rule application only produce very few results. Thus, it is acceptable to restrict the number of iterations to a small constant by allowing a small ratio of loss in the current round of incremental reasoning. Furthermore, since expanding knowledge graph by adding new triples is usually a continuous process, the triples not being deduced by the previous round of incremental reasoning can still be deduced by the following rounds of incremental reasoning through carefully designed process as shown in Fig. 3. This incremental reasoning process starts with an initial knowledge graph K_0 whose theory closure K'_1 is obtained by a normal full reasoning algorithm such as KGRL. Then when an incremental expansion T_1 to K_0 occurs, an incremental reasoning is performed by applying

OWL2 RL rules according to the approaches proposed above for a constant iteration with K'_1 and T_1 as inputs. Triples in T_1 are merged with K'_1 to form K'_2 since triples in T_1 is not new now. All new triples deduced during the incremental reasoning are put together as N_1 which shall be merged with addition triples T_2 in next incremental expansion T_2 to $K'_2 = K'_1 \cup T_1$. After another round of incremental reasoning with K'_2 and $N_1 \cup T_2$ as inputs, another set N_2 of deduced new triples are obtained and $K'_3 = K'_2 \cup N_1 \cup T_2$ is formed as the set of all triples (deduced or not) which have been used for reasoning. This process continues whenever there is an expansion to the knowledge graph K_n .

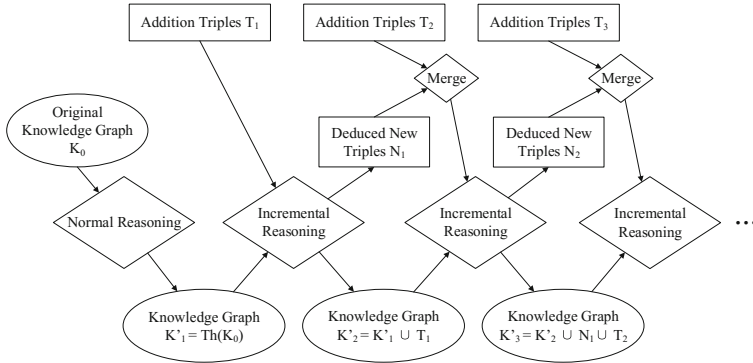


Fig. 3. The expanding and incremental reasoning process of knowledge graph

It can be observed that, in this process, the loss in the previous round of incremental reasoning can be made up by the following rounds, which helps controlling the total loss of each round of incremental reasoning. Finally, the completeness of the final results can still be guaranteed by allowing extra rounds of rule application after the final addition of triples to the knowledge graph.

Based on the above discussion, we can design the following algorithms for filtering irrelevant triples and incremental reasoning.

Irrelevant Triples Filtering Algorithm. The irrelevant triples filtering algorithm mainly relies on the $T_i/N_{i-1} \cup T_i$ ($i \geq 1$), the ontology \mathcal{O} of the knowledge graph, and the patterns of the premise of the eight rules described in Table 1. Algorithm 1 demonstrates the design of filtering algorithms by using the prp-fp rule as an example. For the irrelevant triples filtering of prp-fp rule, we firstly finding all properties of type “owl:FunctionalProperty” in the ontology of knowledge graph. Then we find the pattern in the premise of prp-fp rule that two triples have the same s and p , i.e. start from the same entity and connected through the same property. Based on this pattern, we can find all (s, p) pairs in the set T'_i of new triples which come from triples contain properties p of type “owl:FunctionalProperty”. Finally, all these (s, p) pairs are used to filter out all

Algorithm 1. The irrelevant triples filtering algorithm for the prp-fp rule

Input: \mathcal{O} : the ontology; $T'_1 = T_1$, $T'_i = N_{i-1} \cup T_i$ ($i \geq 1$): new triples; K'_i : old triples

Output: Remaining triples in K'_i after filter out all irrelevant triples

```

1: function FILTER_PRP-FP( $T'_i, K'_i$ )
2:    $\mathcal{P}_{fp} = \{p \mid (p, \text{rdf} : \text{type}, \text{owl} : \text{FunctionalProperty}) \in \mathcal{O}\}$ 
3:    $\mathcal{P}_N = \{(s, p) \mid p \in \mathcal{P}_{fp} \text{ and } (s, p, o) \in T'_i\}$ 
4:    $\text{res} = \{(s, p, o) \mid (s, p) \in \mathcal{P}_N \text{ and } (s, p, o) \in K'_i\}$ 
5:   return res
6: end function

```

irrelevant triples in the large set of old triples K'_i . In fact, different filter conditions can be designed for the same rule, but strict conditions always cost more time for a large scale knowledge graph.

The Main Incremental Reasoning Algorithm. The main incremental reasoning is designed as the following Algorithm 2 based the above discussion. Here the number of rule application iteration is set to 1, because for most of the tested datasets, a single iteration is enough to control the loss.

Algorithm 2. Incremental reasoning algorithm for knowledge graph

Input: $K'_i; T_i; N_{i-1}; \mathcal{R}_M; \mathcal{R}; M$

Output: $K'_{i+1}; N_i$

```

1: function RULEAPPLY(rule, Tr)
2:   Applying rule to a set Tr of triples
3:   return deduced triples
4: end function
5:  $T'_i = T_i \cup N_{i-1}$ 
6:  $N_i = \emptyset$ 
7: for rule  $\in \mathcal{R}$  do
8:   if rule  $\in \mathcal{R}_M$  then
9:     res = RuleApply(rule, Filter_rule( $T'_i, K'_i$ )  $\cup T'_i$ )
10:  else
11:    res = RuleApply(rule,  $T'_i$ )
12:  end if
13:   $N_i = N_i \cup \text{res}$ 
14: end for
15:  $K'_{i+1} = K'_i \cup T'_i$ 
16:  $N_i = N_i - K'_{i+1}$ 

```

4 Experiments and Evaluations

All experiments are conducted in a Spark cluster with Spark 1.6.0, Hadoop 2.6, and JDK 1.8.0. This cluster consists of 12 machines with Intel(R) Xeon(R) E5-4607 v2 CPU, 32 GB memory, and Ubuntu Linux 12.04 (64bit) operating system.

The datasets used in the experiments are English version of DBpedia [10] April 2015 dataset and LUBM dataset [7]. For DBpedia dataset, we remove tags and profiles of entities, links linking to external datasets such as Yago, and links linking to home pages of other web sites. These data are removed because they are not relevant to our reasoning algorithm and consume too much memory. The DBpedia dataset used for experiments contains 107,622,682 triples. For LUBM dataset, we use its official data generation tool, UBA, to generate a knowledge graph of 500 universities, which contains 69,099,760 triples.

4.1 Time Comparison

In order to compare the time consumption of the incremental reasoning and the normal reasoning algorithm KGRL, we randomly extract certain number ($10^2, 10^3, 10^4, 10^5, 10^6$ respectively) of triples from the DBpedia dataset as new triples for incrementally adding back to the knowledge graph. Then we run the incremental reasoning algorithm to incrementally update the theory closure of the original knowledge graph and KGRL algorithm for theory closure reasoning of the expanded knowledge graph ten times for each number.

The runtime data of both reasoning algorithms are listed in Table 2, with T_n be the runtime of normal reasoning with KGRL and T_i be the runtime of the proposed incremental reasoning algorithm. And Fig. 4 presents the average runtime of the two algorithms. As can be seen, the proposed incremental reasoning algorithm is much faster than KGRL. This advantage keeps obvious even when the triples used for incremental expansion increases to a relatively big size. The experiment results also show that when number of the triples is less than 10^5 , a full theory closure reasoning on the updated knowledge graph takes almost 10

Table 2. Runtime comparison between KGRL and incremental reasoning algorithms

Time (s)		Number of addition triples									
		10^2		10^3		10^4		10^5		10^6	
		T_n	T_i	T_n	T_i	T_n	T_i	T_n	T_i	T_n	T_i
Groups	1	4062	335	5362	381	4608	401	4973	464	4420	1027
	2	3378	373	5570	375	4244	406	4091	510	4640	1027
	3	2805	327	6362	385	4413	392	4376	478	5781	1017
	4	2721	283	6093	344	6992	391	4329	460	4925	1017
	5	2755	293	6319	382	4283	408	4149	466	4704	991
	6	3012	290	5457	382	3830	395	6953	453	4755	1017
	7	3427	355	5851	391	7156	397	4239	526	4884	1015
	8	4036	331	5680	390	4650	399	4070	457	4772	986
	9	2506	320	6004	376	4752	410	4111	471	4754	995
	10	2658	365	6031	390	4831	405	4663	456	4982	1011
Average		3136	327	5872	380	4975	400	4495	474	4759	1010

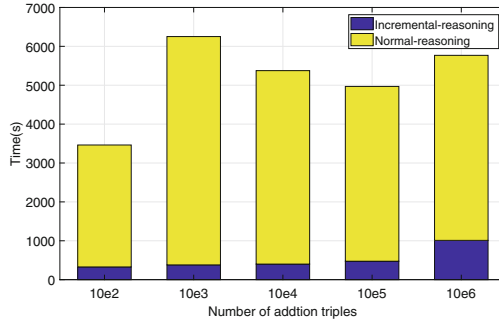


Fig. 4. Comparison over average time

times longer than an incremental reasoning base on the original theory closure and addition triples. When the number increased to 10^6 , a full theory closure reasoning still takes almost quadruple the time incremental reasoning takes. The point where the advantage become less obvious apparently is determined by the computational power of the cluster. The more powerful the computing cluster is, the later the turning point comes.

From Fig. 4, we can also see that the time used by the KGRL full reasoning method fluctuates dramatically, especially when the number of addition triples increases to 10^3 . This may because of the number of triples used for reasoning is very huge, almost 180,000,000. So the network condition and the initial distribution of data in different nodes can greatly affect the total runtime of reasoning.

4.2 Loss Evaluation

Our strategy for only preserving relative completeness of the results for each round of incremental reasoning can prevent loss of deduced triples. In order to evaluate its effectiveness, we simulate a continuous incremental reasoning process to observe the trend of loss. We extract 500,000 triples from the testing knowledge graph, and then split the extraction data into 10 sets and add back one set per round. We compare the number of final triples obtained by incremental reasoning algorithm with the results of a full theory closure reasoning by using KGRL. We conduct this experiment on both DBpedia and LUBM.

Figures 5 and 6 show the number of triples obtained by two reasoning algorithms in a continuous expanding process of 10 rounds. The experiments show that no result loss under both the DBpedia and LUBM datasets. Figure 5 is nearly linear because LUBM dataset is generated in a same pattern. In fact, the loss is affected by the complexity of the knowledge graph and its corresponding ontology. So for common knowledge graphs with relatively simple ontologies, the loss would keep minimum.

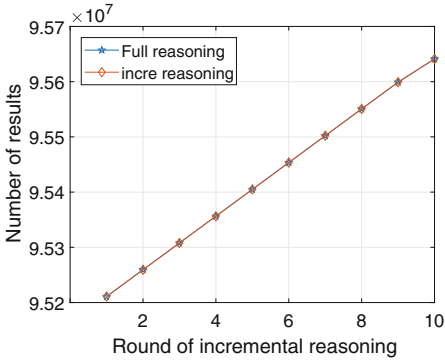


Fig. 5. LUBM: trend of loss

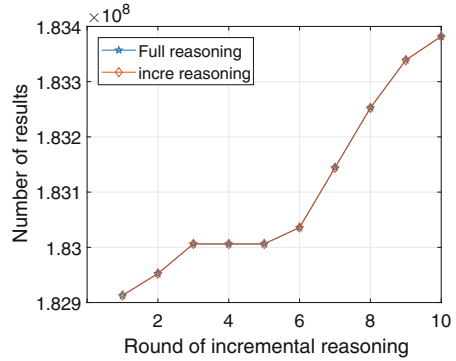


Fig. 6. DBpedia: trend of loss

5 Conclusion

In this paper, an incremental reasoning algorithm for large scale knowledge graph is proposed based on previous work on rule-based distributed reasoning algorithm KGRL [17]. When a small number of new triples are added to a large-scale knowledge graph, the algorithm can effectively perform an incremental update to the original reasoning results and avoid performing a full reasoning over the entire new knowledge graph. In this paper, we analyze the challenges of incremental reasoning and proposed two approaches for designing an efficient incremental reasoning algorithm. Finally, experiments are conducted on DBpedia [10] April 2015 and LUBM [7] datasets. And the experimental results show that the proposed incremental algorithm is efficient and preserve the relative completeness of the results.

For future work, further improvements to the efficiency will be one of our focuses. As mentioned above, there are still some instabilities of the performance of the reasoning algorithm due to the initial data distribution. It is very important to investigate how the data distribution effect the performance and what is the optimal distribution (or partition) for the input knowledge graph. Besides, the efficiency of current filtering algorithm for irrelevant triples still needs to be improved.

Acknowledgments. This work was supported by National Natural Science Foundation of China (Grand No. 61502022) and State Key Laboratory of Software Development Environment (Grand No. SKLSDE-2017ZX-17).

References

1. Anicic, D., Fodor, P., Rudolph, S., Stojanovic, N.: EP-SPARQL: a unified language for event processing and stream reasoning. In: WWW 2011, pp. 635–644. ACM (2011)
2. Bazoobandi, H.R., Beck, H., Urbani, J.: Expressive stream reasoning with laser. CoRR abs/1707.08876 (2017)

3. Beck, H., Dao-Tran, M., Eiter, T., Fink, M.: LARS: a logic-based framework for analyzing reasoning over streams. In: Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, AAAI 2015, pp. 1431–1438. AAAI Press (2015)
4. Broekstra, J., Kampman, A., van Harmelen, F.: Sesame: a generic architecture for storing and querying RDF and RDF schema. In: Horrocks, I., Hendler, J. (eds.) ISWC 2002. LNCS, vol. 2342, pp. 54–68. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-48005-6_7
5. Dean, J., Ghemawat, S.: MapReduce: simplified data processing on large clusters. In: Proceedings of the 6th Conference on Symposium on Operating Systems Design and Implementation, OSDI 2004, vol. 6, p. 10. USENIX Association, Berkeley (2004)
6. Gu, R., Wang, S., Wang, F., Yuan, C., Huang, Y.: Cichlid: efficient large scale RDFS/OWL reasoning with spark. In: IEEE International Parallel and Distributed Processing Symposium, pp. 700–709 (2015)
7. Guo, Y., Pan, Z., Heflin, J.: LUBM: a benchmark for OWL knowledge base systems. *J. Web Sem.* **3**(2–3), 158–182 (2005)
8. Kim, J., Park, Y.: Scalable OWL-horst ontology reasoning using SPARK. In: 2015 International Conference on Big Data and Smart Computing (BIGCOMP), pp. 79–86, February 2015
9. Kimmig, A., Bach, S.H., Broecheler, M., Huang, B., Getoor, L.: A short introduction to probabilistic soft logic. In: Proceedings of the NIPS Workshop on Probabilistic Programming: Foundations and Applications, pp. 1–4 (2012)
10. Lehmann, J., et al.: DBpedia - a crystallization point for the web of data. *J. Web Semant.* **7**(3), 154–165 (2009)
11. McBride, B.: Jena: a semantic web toolkit. *IEEE Internet Comput.* **6**(6), 55–59 (2002)
12. Parsia, B., Matentzoglou, N., Gonçalves, R.S., Glimm, B., Steigmiller, A.: The OWL reasoner evaluation (ORE) 2015 competition report. *J. Autom. Reason.* **59**(4), 455–482 (2017)
13. Sirin, E., Parsia, B., Grau, B., Kalyanpur, A., Katz, Y.: Pellet: a practical OWL-DL reasoner. *Web Semant. Sci. Serv. Agents World Wide Web* **5**(2), 51–53 (2007)
14. Tommasini, R., Della Valle, E., Mauri, A., Brambilla, M.: RSPLab: RDF stream processing benchmarking made easy. In: d’Amato, C., et al. (eds.) ISWC 2017. LNCS, vol. 10588, pp. 202–209. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-68204-4_21
15. Urbani, J., Kotoulas, S., Maassen, J., van Harmelen, F., Bal, H.: OWL reasoning with WebPIE: calculating the closure of 100 billion triples. In: Aroyo, L., et al. (eds.) ESWC 2010. LNCS, vol. 6088, pp. 213–227. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13486-9_15
16. Urbani, J., Kotoulas, S., Oren, E., van Harmelen, F.: Scalable distributed reasoning using MapReduce. In: Bernstein, A., et al. (eds.) ISWC 2009. LNCS, vol. 5823, pp. 634–649. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-04930-9_40
17. Wei, Y., Luo, J., Xie, H.: KGRL: an OWL2 RL reasoning system for large scale knowledge graph. In: 12th International Conference on Semantics, Knowledge and Grids, SKG 2016, Beijing, China, 15–17 August 2016, pp. 83–89 (2016)
18. Zaharia, M., Chowdhury, M., Franklin, M.J., Shenker, S., Stoica, I.: Spark: cluster computing with working sets. In: Proceedings of the 2nd USENIX Conference on Hot Topics in Cloud Computing, HotCloud 2010, p. 10. USENIX Association, Berkeley (2010)



Relation Classification Using Coarse and Fine-Grained Networks with SDP Supervised Key Words Selection

Yiping Sun^{1,2}, Yu Cui^{1,2}, Jinglu Hu², and Weijia Jia^{1,3}(✉)

¹ School of Electronic Information and Electrical Engineering,
Shanghai JiaoTong University, Shanghai, China

{sunacc, adacui523}@sjtu.edu.cn, jia-wj@cs.sjtu.edu.cn

² Graduate School of Information, Production and Systems, Waseda University,
Kitakyushu-shi, Japan
jinglu@waseda.jp

³ University of Macau, Macau, China

Abstract. In relation classification, previous work focused on either whole sentence or key words, meeting problems when sentence contains noise or key words are extracted falsely. In this paper, we propose coarse and fine-grained networks for relation classification, which combine sentence and key words together to be more robust. Then, we propose a word selection network under shortest dependency path (SDP) supervision to select key words automatically instead of pre-processed key words and attention, which guides word selection network to a better feature space. A novel opposite loss is also proposed by pushing useful information in unselected words back to selected ones. In SemEval-2010 Task 8, results show that under the same features, proposed method outperforms state-of-the-art methods for relation classification.

Keywords: Relation classification · Coarse and fine-grained networks
Key words selection · Shortest dependency path · Opposite loss

1 Introduction

Relation classification has been an efficient tool in understanding natural language. Many applications, like knowledge base completion [5] and question answering [7], have been developed based on it. Relation classification can be described as follows: Given a sentence $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n$ with two entities $\mathbf{e}_1, \mathbf{e}_2$, the model should classify the relation \mathbf{y} of two entities where relation is defined by dataset. For example: *A woman_{e1} has been placed into the house_{e2} as well.* In this sentence, entity woman and house have the relation of Entity-Destination (e_1, e_2) .

In relation classification, previous work focused on either whole sentence or key words. [8, 11] introduced sophisticated models to extract features from whole sentence. But the performance may be affected by noise from unrelated words.

Other works concentrated on key words, which are difficult to extract. [3,12] introduced attention mechanism to extract key words based on classification loss, facing problems when sentence is complicated. [2,9] directly used pre-processed key words like SDP as input, which brings noise as well because SDP may not be correct and sometimes cannot be extracted.

In this paper, we propose coarse and fine-grained networks for relation classification, which of inputs are whole sentence (coarse) and selected key words (fine-grained) separately so that sentence and key words can complement each other when sentence contains too much noise or key words are not extracted correctly. Then, we propose word selection network under SDP supervision to select key words automatically instead of attention and pre-processed key words. Word selection network gets high-level features of coarse network as input and generates selection weight for each word as input of fine-grained network. We propose SDP as a supervision signal to guide word selection network. SDP can give valuable information for key words selection for two reasons: First, SDP can represent grammar dependency between two entities which is suitable for relation classification. Second, various literature is proposed to extract dependency parser tree so we do not need to label SDP despite the difficulty there is no labeled data representing which words should be selected or not. In addition, we propose opposite loss to enhance the word selection network by pushing useful information in unselected words back to selected words.

2 Methodology

2.1 The Overview of Model Architecture

- Input Representation: Words are embedded into word and position embedding, the latter is based on the relative positions of two entities.
- Coarse and Fine-Grained Relation Classification: The first network accepts the whole sentence as coarse feature and the second network accepts selected words as fine-grained feature.
- Key Words Selection: For each word, word selection network generates a selection weight representing whether it should be selected or not. We design SDP supervision loss L_{sdp} , Classification loss L_{cls} , Opposite loss L_{oppo} to guide word selection network jointly.
- Output Layer: Classification results given by two classification networks are combined together (Fig. 1).

2.2 Input Representation

The input representation is composed of two parts, word embedding and position embedding. For word embedding, each word x_i is mapped to a high-dimensional vector [1] by looking up $WV^{emb} \in R^{d_e \times |V|}$. For position embedding, to highlight two entities in the sentence, relation classification further embeds each word into position embedding vectors [10]. For each word x_i , it will have two relative

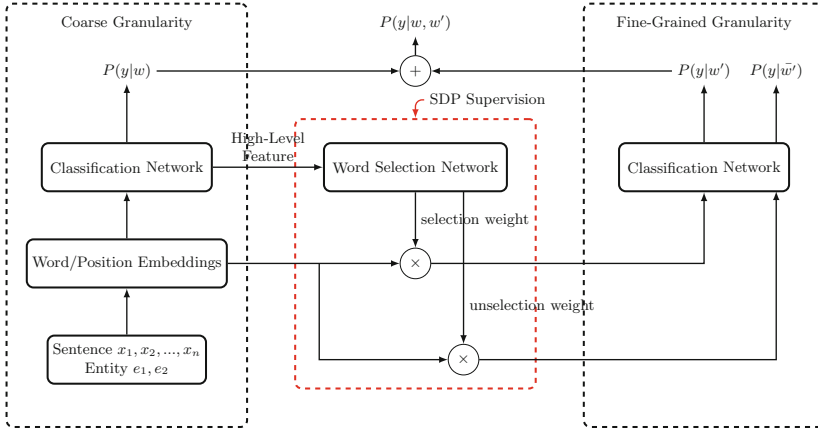


Fig. 1. Overview of Coarse and Fine-Grained Networks with SDP Supervised Key Words Selection (SDP-CFN)

position indexes, p_i^1 and p_i^2 , of two entities e_1 and e_2 . Then each position index is mapped to a high-dimensional vector by looking up $PV^{emb} \in R^{d_p \times |P|}$. Finally, the overall input representation for each word is

$$w_i = [(wv_i)^T, (pv_i^1)^T, (pv_i^2)^T]^T \tag{1}$$

2.3 Coarse and Fine-Grained Relation Classification

In this section, we introduce the coarse and fine-grained relation classification mechanism. The first network accepts coarse feature which is the whole sentence and the second network accepts fine-grained feature which is the key words. We combine two classification networks together for two reasons. First, two networks can complement each other. Second, coarse network can give high-level features to word selection network and fine-grained network can also guide word selection network. Since our contribution is independent from specific classification network architecture, our method is a model-free method, we can change the classification network to any effective model. After two classification networks get the probability, we add them together to get the final result.

$$y_{pred} = argmax_y P(y|w; W, B) + P(y|w'; W', B') \tag{2}$$

$$w' = w \times sw \tag{3}$$

where sw is the selection weight generated by word selection network, W, B and W', B' is the weight and bias of two networks independently.

Bi-LSTM with Entity Pair Attention. For the specific structure of each classification network, we employ Bi-LSTM with entity pair attention [3]. The main idea is to generate the self-learned attention vector using entity

pair information instead of randomly generated vector. After embeddings pass through Bi-LSTM cells, hidden state $H_i = [h_i^f, h_i^b]$ is combined with two directional hidden states $h_i^f, h_i^b \in R^{d_h}$. Attention vector $h_e \in R^{2d_h}$ is the last hidden state generated by a unidirectional LSTM using entity pair information. By multiplying each hidden state with attention vector, we can get the attention weight α for each word.

$$\alpha_i = \frac{\exp(H_i^T h_e)}{\sum_i \exp(H_i^T h_e)} \quad (4)$$

Finally, the final sentence representation is the sum of hidden states adjusted by attention weight and classifier is the softmax function.

$$s = \sum_i \alpha_i H_i \quad (5)$$

$$p(y|w) = \text{softmax}(W_c s + b_c) \quad (6)$$

$$L_{cls}(P(y|w), \hat{y}) = - \sum_{i=1}^N \log(P(\hat{y}^{(i)} | w^{(i)})) \quad (7)$$

2.4 Key Words Selection

Structure. We realize key words selection mechanism by multiplying a selection weight on embedding of each word in Eq. (3). The word selection network aims to generate selection weight for each word.

$$M = BiLSTM(H) \quad (8)$$

$$sw = \sigma(W_{wsn} M + b_{wsn}) \quad (9)$$

Word selection network accepts inputs from hidden states of the previous classification network so that key words selection can learn based on high-level features of relation classification. After passing through a Bi-directional LSTM, two fully connected layers and sigmoid function σ , word selection network generates selection weight $sw \in (0, 1)$ for each word.

Shortest Dependency Path Supervision. If we do not intervene the word selection network and let it train by itself, it is the same as self-learned attention which only tries to lower the loss and turns to be challenging: classification

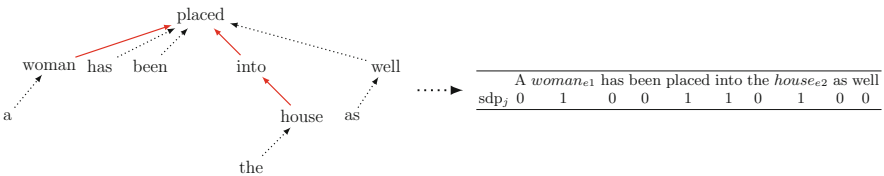


Fig. 2. Example of constructing shortest dependency path supervision

network may be trained on bad selected words and word selection network may also be trained on bad classification network’s feedback.

But in relation classification task, we do not have labeled data to represent which words should be selected or not, which brings huge difficulties. So to solve the above problems and guide the generation of selection weight, we first accept the assumption that words on this path are key words while other are not and ignore the fact that some paths cannot be extracted and incorrect even if extracted. Based on this assumption, we have a valuable supervision signal to represent which words should be selected or not. That is, for words on shortest dependency path, we set their selection weight as 1 and otherwise 0. We employ it as teacher signal and use neg log-likelihood to supervise word selection network (Fig. 2).

$$L_{sdp} = - \sum_{i=1}^N \sum_{j=1}^L sdp_j^{(i)} \log(sw_j^{(i)}) + (1 - sdp_j^{(i)}) \log(1 - sw_j^{(i)}) \quad (10)$$

Opposite Loss. Now word selection network is decided by SDP supervision loss and Classification loss. Since SDP supervision also brings huge noise, we propose opposite loss to enhance the classification loss. The main idea of opposite loss is mainly coming from the fact that for unselected words, if classification network can give a little probability on assigned class (say 0.2), which means in these unselected words, there still remains some words containing valuable information for this relation, so our goal is to push such valuable information back to selected words. Based on it, our realization is to let this probability tend to be 0. Cross-entropy loss is employed to reach the goal.

$$L_{oppo} = - \sum_{i=1}^N \log(1 - p(\hat{y}^{(i)} | \bar{w}'^{(i)})) \quad (11)$$

$$\bar{w}' = w' \times (1 - sw) \quad (12)$$

where \hat{y} is the ground truth, w' , \bar{w}' is the embedding of selected and unselected words, sw is the selection weight.

End-to-End Joint Training. To train the model end-to-end jointly, we first train the model based on SDP supervision loss and classification loss, then transiting to opposite loss by introducing decay rate γ .

$$J = L_{cls}(P(y|w), \hat{y}) + L_{cls}(P(y|w'), \hat{y}) + \gamma^{iter} * L_{sdp} + (1 - \gamma^{iter}) * L_{oppo} \quad (13)$$

3 Experiments

3.1 Dataset, Experimental Setup and Results

Experiments are done on SemEval-2010 Task 8 dataset, which has 10717 sentences, including 8000 training examples and 2717 testing examples. It has 9 actual relation classes and an additional class *other*, indicating entity pair not

Table 1. Comparison with other work reported on SemEval 2010 Task 8

Model	Additional information	F1
SVM [4]	POS, prefixes, morphological, WordNet, dependency parse, Levin classed, ProBank, FrameNet, NomLex-Plus, Google n-gram, paraphrases, TextRunner	82.2
SDP-LSTM [9]	Word/Position embedding+POS+GR+WordNet	83.7
BLSTM [11]	Word/Position embedding+PF+POS+NER+WNSYN+DEP	84.3
depLCNN [6]	Word/Position embedding+Wordnet+words around nominals +Negative sampling from NYT dataset	83.7 85.6
DRNNs [8]	Word/Position embedding+POS+GR+Wordnet +Data augmentation	84.2 86.1
SPTree [2]	Word embeddings+DEP +Wordnet	84.4 85.5
Att-BiLSTM [12]	Word embedding+Position Indicator	84.0
EAtt-BiGRU [3]	Word/Position embedding	84.7
Proposed method	Word/Position embedding+DEP +Wordnet+POS	85.7 86.4

belonging to any relation. We employ official scorer (macro F1) as our evaluation metric, which does not consider class *other* and takes the directionality into consideration. We use GoogleNews-vectors-negative300 for word embedding. The dimension of word vector d_e , position vector d_p and LSTM hidden state d_h is 300, 12, 100 respectively. Decay rate γ for transition is 0.95. The model is optimized by Adadelta with learning rate 1.0 and batch size is 20. To overcome the overfitting problem, we apply dropout on embedding layer and ultimate layer with rate 0.5 and L2 regularization with strength of 1×10^{-5} .

Table 1 shows comparison between proposed method and other state-of-the-art systems.

- SVM [4]: The first top traditional method by equipping plenty of features.
- SDP-LSTM [9]: Treated shortest dependency path as LSTM’s input directly. Our baseline from fine-grained network’s perspective.
- DepLCNN [6]: Not only equipped shortest dependency path in CNN but also improved result by negative sampling from NYT dataset.
- DRNNs [8]: Deep recurrent neural networks with data augmentation.
- SPTree [2]: Joint extraction of entities and relations under tree LSTM.
- BLSTM [11]: Bidirectional LSTM to extract high-level features.
- Att-BiLSTM [8]: First introduced attention into relation classification.
- EAtt-BiGRU [3]: Further improved attention mechanism by generating attention weight with prior knowledge of entity pair information. Our baseline from coarse network’s perspective.

Without other features, proposed method gets the highest F1 score of 85.7%. With high-level features, result has been improved to 86.4%. The improvement of results comes from two parts. First, instead of either using coarse or fine-grained network, proposed method combines them together to make classification

Table 2. Comparison of word selection network with attention and SDP supervision

Example 1	
Attention	The school master _{e1} teaches the lesson with a stick _{e2}
SDP Supervision	The school master_{e1} teaches the lesson with a stick_{e2}
Proposed Method	The school master_{e1} teaches the lesson with a stick_{e2}
Example 2	
Attention	The magazine _{e1} was founded in order to keep athletes serving as soldiers informed about their sport _{e2} back home
SDP Supervision	The magazine_{e1} was founded in order to keep athletes servng as soldiers informed about their sport_{e2} back home
Proposed Method	The magazine_{e1} was founded in order to keep athletes serving as soldiers informed about their sport_{e2} back home
Example 3	
Attention	In 1952 when Fidel ran for congressman _{e1} on the Ortodoxo party _{e2} ticket _{e2} it was Father who helped finance his campaign
SDP Supervision	In 1952 when Fidel ran for congressman_{e1} on the Ortodoxo party_{e2} ticket_{e2} it was Father who helped finance his campaign
Proposed Method	In 1952 when Fidel ran for congressman_{e1} on the Ortodoxo party_{e2} ticket_{e2} it was Father who helped finance his campaign
Example 4	
Attention	The synthesist is thus carrying out a role which is analogous to that of a player _{e1} in an orchestra _{e2}
SDP Supervision	The synthesist is thus carrying out a role which is analogous to that of a player_{e1} in an orchestra_{e2}
Proposed Method	The synthesist is thus carrying out a role which is analogous to that of a player_{e1} in an orchestra_{e2}

Table 3. Precision of two classification network on M1: proposed method, M2: remove opposite loss on M1, M3: remove word selection network on M2

		M1	M2	M3
Same opinion		74.6%	73.1%	72.4%
Different opinion	Classification I right	3.7%	4.1%	4.3%
	Classification II right	3.9%	4.6%	4.7%
Overall		82.2%	81.8%	81.4%

have better generalization ability. Second, word selection network under SDP supervision guides more robust key words selection and connects two networks in a joint model, thus improving results more.

Detailed Analysis. To further analyze the effectiveness of proposed method, we compare word selection network with attention [3] and SDP supervision. In Table 2, Example 1 shows that under easy sentence, proposed word selection network, SDP supervision and attention can all choose the correct key words. However in Example 2, SDP supervision cannot distinguish useless words between entities but proposed method and attention can distinguish them. In Example 3, proposed method further extracts more (*ran for*) than attention and SDP supervision, which will play an important role for relation classification, because only one key word (*on*) is still difficult for classifying. In Example 4, attention cannot effectively extract correct key words, which we find such phenomenon is common in dataset.

Then we analyze the precision of two classification networks on removing each component in Table 3. Without any proposed method, two networks have 72.4% agreement while with proposed word selection network, the agreement has been improved to 74.6%, which means under the co-training framework, two classification networks start to extend to the same feature space.

4 Conclusion

In this paper, we propose coarse and fine-grained networks for relation classification, which combine sentence and key words together and are more robust than either coarse network or fine-grained network. Then, to select key words efficiently, we propose a word selection network under shortest dependency path (SDP) supervision instead of attention and pre-processed key words, which guides word selection network to a better feature space and successfully overcomes the difficulty lack of labeled data representing which words should be selected. Results on Semeval 2010 Task 8 show that under the same features, proposed method outperforms state-of-the-art methods for relation classification. A further detailed analysis proves that the proposed method selects more robust key words than self-learned attention and SDP supervision. Analysis also shows that two classification networks are extending to the same feature space due to connection of word selection network.

Acknowledgements. This work is supported by FDCT 0007/2018/A1, DCT-MoST Joint-project No. (025/2015/AMJ) of SAR Macau; University of Macau Funds Nos: CPG2018-00032-FST & SRG2018-00111-FST; Chinese National Research Fund (NSFC) Key Project No. 61532013; National China 973 Project No. 2015CB352401 and 985 Project of Shanghai Jiao Tong University: WF220103001. We also thank Xinsong ZHANG, Lester James V. Miranda and Mingyang YU for revising this paper.

References

1. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: NIPS, pp. 3111–3119 (2013)
2. Miwa, M., Bansal, M.: End-to-end relation extraction using LSTMs on sequences and tree structures. In: Proceedings of ACL, vol. 1, pp. 1105–1116 (2016)
3. Qin, P., Xu, W., Guo, J.: Designing an adaptive attention mechanism for relation classification. In: IJCNN, pp. 4356–4362. IEEE (2017)
4. Rink, B., Harabagiu, S.: UTD: classifying semantic relations by combining lexical and semantic resources. In: Proceedings of SemEval, pp. 256–259 (2010)
5. Socher, R., Chen, D., Manning, C.D., Ng, A.: Reasoning with neural tensor networks for knowledge base completion. In: NIPS, pp. 926–934 (2013)
6. Xu, K., Feng, Y., Huang, S., Zhao, D.: Semantic relation classification via convolutional neural networks with simple negative sampling. In: Proceedings of EMNLP, pp. 536–540 (2015)
7. Xu, K., Reddy, S., Feng, Y., Huang, S., Zhao, D.: Question answering on freebase via relation extraction and textual evidence. In: Proceedings of ACL, vol. 1, pp. 2326–2336 (2016)
8. Xu, Y., et al.: Improved relation classification by deep recurrent neural networks with data augmentation. In: Proceedings of COLING, pp. 1461–1470 (2016)
9. Xu, Y., Mou, L., Li, G., Chen, Y., Peng, H., Jin, Z.: Classifying relations via long short term memory networks along shortest dependency paths. In: Proceedings of EMNLP, pp. 1785–1794 (2015)

10. Zeng, D., Liu, K., Lai, S., Zhou, G., Zhao, J.: Relation classification via convolutional deep neural network. In: Proceedings of COLING, pp. 2335–2344 (2014)
11. Zhang, S., Zheng, D., Hu, X., Yang, M.: Bidirectional long short-term memory networks for relation classification. In: Proceedings of PACLIC, pp. 73–78 (2015)
12. Zhou, P., et al.: Attention-based bidirectional long short-term memory networks for relation classification. In: Proceedings of ACL, vol. 2, 207–212 (2016)

Author Index

- Alallaq, Noora I-238
Alanoz, Qusay I-238
Al-Azzawi, Adil I-238
Al-khiza'ay, Muhmmad I-238
- Bai, Qingchun I-260
Bi, Jingping I-287
Bindra, Simran Kaur I-337
- Cai, Dunbo II-69
Cao, Jinxin II-184
Cao, Yanan I-491
Chen, Bingguang I-351
Chen, Enhong I-187
Chen, Hechang II-425
Chen, Hongchao I-16
Chen, Hua II-396
Chen, Jiahui I-169
Chen, Juan I-351, II-266
Chen, Mengwei I-260
Chen, Nengcheng I-148
Chen, Rong II-36
Chen, Shizhan II-324, II-383
Chen, Xiaofang II-3
Chen, Yan I-187
Cheng, Shuzhi I-125
Cui, Jiayu II-24
Cui, Yu I-514
- Dang, Jianwu II-184
Di, Haibo II-350
Ding, Dongtai II-413
Douligeris, Christos I-442
Duan, Hua I-415
Duan, Mingyue I-469
- E, Xu I-268
- Fan, Xinxin I-287
Fang, Shancheng II-197
Feng, Zhiyong II-324, II-383
Fogelman-Soulié, Françoise II-383
Fu, Wenjing I-324
Fu, Zhao II-3
- Gabrys, Bogdan I-362
Gai, Lei II-117
Gao, Chao II-438
Gao, Jian II-36
Gao, Jie II-350
Gao, Linan I-42
Gao, Mengyu II-144
Gao, Wei II-69
Gao, Yuhui II-69
Ge, Ce I-160
Ge, Yong I-187
Goel, Vipra I-337
Gu, Ming II-337
Gu, Shunhua II-477
Gu, Xiwu I-55, I-461
Guo, Jingyi II-452
Guo, Li I-213, I-491
Guo, Shanqing I-42
Guo, Shikai II-36
- Habimana, Olivier I-461
Hang, Zhi I-433
Hao, Yuwei I-452
He, Ben I-104
He, Dongxiao I-362, II-324, II-383
He, Fengling I-452
He, Jun II-244
He, Liang I-260
He, Ming I-187
He, Ruifang II-413, II-465
He, Tingnian II-313
Hong, Xiaoguang I-324
Honkisz, Krzysztof I-91
Hou, Jian I-268
Howard, Catherine II-132
Hu, Gang I-433
Hu, He II-291
Hu, Jing II-274
Hu, Jinglu I-514
Hu, Junfeng II-209
Hu, Qinmin I-260
Hu, Xiaoli II-324
Hu, Yupeng I-324
Huang, Jing I-469

- Huang, Keman II-324
 Huang, Lin II-283
 Huang, Weiqing II-221
 Hung, Kwok-Wai II-255
 Huo, Qiming I-160
- Jia, Haiyang I-351, II-266
 Jia, Weijia I-514
 Jia, Zhaocong II-89
 Jiang, Jianguo II-221
 Jiang, Jianmin II-255
 Jiang, Jing I-3
 Jiang, Zhengang I-276
 Jin, Di I-362, II-184
 Jin, Xiaolong II-371
 Jing, Quanliang I-287
- Kalogeraki, Eleni Maria I-442
 Kang, WenJie I-433
 Ke, Wei I-148, I-169, I-177
 Khan, Aamir I-388, II-209
 Kluza, Krzysztof I-91
- Lei, Kai I-479
 Li, Chao I-169, I-177
 Li, Chunfang I-117
 Li, Gang I-200, II-221
 Li, Junfan II-301
 Li, Li II-244
 Li, Ruixuan I-55, I-461
 Li, Xianghua II-452
 Li, Xin I-187
 Li, Yande I-388
 Li, Yuan I-311
 Li, Yuhua I-55, I-461
 Li, Zhanshan II-59
 Li, Zhaokui II-283
 Li, Zhe II-59
 Liang, Shining I-299
 Liao, Lejian I-3
 Liao, Shizhong II-301, II-337
 Lin, Jianfeng II-274
 Lin, Zuoquan II-89
 Liu, Chao II-221
 Liu, Chengmei II-438
 Liu, Cong I-415
 Liu, Cuiwei II-283
 Liu, Dong I-137
 Liu, Guiquan I-187, II-274
 Liu, Hao I-491
- Liu, Jiayun I-377
 Liu, Jiming II-425
 Liu, Jingshuang I-67
 Liu, Lei I-177
 Liu, Li I-388, II-209
 Liu, Lichao I-187
 Liu, Liting I-125
 Liu, Meng II-48
 Liu, Minghao II-12
 Liu, Qi I-187
 Liu, Shuang I-250
 Liu, Xin I-433
 Liu, Xueyan II-396
 Liu, Yanbing I-213, I-491
 Liu, Ziyang I-362
 Liu, Ziyu I-469
 Lou, Chao II-350
 Lu, Yihong I-117
 Lu, Yue II-221
 Lu, Yuhang I-213
 Luo, Jie I-503
 Lv, Chengcong I-268
 Lv, Jianming II-171
 Lv, Kai I-177
- Ma, Feifei II-12
 Ma, Huifang II-313
 Ma, Wenchao II-209
 Ma, Zhiyuan II-159
 Maheswari, N. I-238
 Mayer, Wolfgang II-132
 Mu, Kedian I-311
 Musial, Katarzyna I-362
- Nagpal, Sushama I-337
 Ni, Jingcheng II-266
 Nie, Peng I-29
 Niu, Zhendong II-232
 Nyamawe, Ally S. II-232
- Ouyang, Dantong II-48, II-144
 Ouyang, Yuanxin I-79
- Panayiotopoulos, Themis I-442
 Papastergiou, Spyridon I-442
 Pei, Yulong I-415
 Peng, Zhaohui I-324
 Philp, Dean II-132
 Polemi, Nineta I-442

- Qi, Qi I-160
 Qian, Chengcheng I-250
 Qian, Kun I-169
 Qiu, Chaoming II-255
 Qiu, Riming I-311
 Qiu, Yunqi II-371

 Ran, Yanhua I-104
 Rong, Wenge I-67, I-79, II-159

 Sekar, Booma Devi I-426
 Shen, Ying I-479
 Sheng, Hao I-148, I-169, I-177
 Shi, Jiahao II-403
 Shi, Libin I-67, II-159
 Shi, Minyong I-117
 Shi, Weili I-276
 Shi, Wenxuan I-125
 Shi, Xiangbin II-283
 Shi, Zhenkun I-299, I-452
 Sikos, Leslie F. II-132
 Song, Wenzhuo II-396
 Song, Yanjing I-491
 Stumptner, Markus II-132
 Su, Bing I-148
 Su, Guoxin I-388
 Sun, Haifeng I-160
 Sun, Nannan II-197
 Sun, Yiping I-514

 Tan, Haining I-287
 Tan, Jianlong I-200, I-213
 Tan, Xu II-396
 Tan, Yuyang II-274
 Tang, Weiqiang I-287
 Tao, Ya II-48
 Tian, Zhengxi I-67
 Tomer, Shruti I-337

 Voigt, Shaun II-132

 Wang, Beibei I-469
 Wang, Danni I-388
 Wang, Haozhao I-461
 Wang, Hengliang I-311
 Wang, Hui I-250, I-426
 Wang, Jianrong II-350
 Wang, Jingyu I-160
 Wang, Jingyuan II-244
 Wang, Jinyan II-78

 Wang, Liangguo I-3
 Wang, Lizhen I-403
 Wang, Meng I-137
 Wang, Qingyue I-491
 Wang, Shaoni I-200
 Wang, Shengsheng I-137, I-377
 Wang, Tao I-79
 Wang, Tengjiao II-117
 Wang, Xiaoming II-117
 Wang, Xintong II-171
 Wang, Xitong II-24
 Wang, Yan II-283
 Wang, Yanmeng II-159
 Wang, Yifei I-503
 Wang, Yixuan II-266
 Wang, Yong II-403
 Wang, Yuanzhuo II-371
 Wang, Zeyu I-42
 Wei, Jiahui II-313
 Wei, Kai I-260
 Wei, Miaomiao II-36
 Wei, Xing II-361
 Wen, Yanlong I-29
 Wiśniewski, Piotr I-91
 Wu, Jingli II-78
 Wu, Yong II-78
 Wu, Yuanyuan II-477
 Wu, Zhongbin II-102
 Wu, Zuoxi I-351

 Xia, Haiyang I-200
 Xiao, Qing I-403
 Xie, Hongtao II-197
 Xie, Meng II-313
 Xin, Yingchu II-438
 Xiong, Zhang I-67, I-79, II-159
 Xu, Jungang I-104
 Xu, Tianyi II-350
 Xu, Xiaofei II-244
 Xu, Xiaolong II-477
 Xu, Yang I-324
 Xu, Zhiqiang I-226
 Xue, Guangtao I-169
 Xue, Shuai I-452

 Yan, Jun II-12
 Yang, Bo I-469, II-361, II-425
 Yang, Chunxue I-311
 Yang, Da I-148

- Yang, Dongbao II-197
Yang, Huamin I-276
Yang, Jing II-403
Yang, Mingqi II-59
Yang, Peizhong I-403
Yang, Xue II-383
Yang, Xuechen I-16
Yao, Sicheng I-351
Ye, Yutong II-78
Ye, Yuxin II-144
Yin, Minghao II-69
Yousif, Abdallah II-232
Yu, Hong II-3
Yu, Jing I-213
Yu, Mei II-350
Yu, Min II-221
Yu, Qiang-yuan I-377
Yu, Ruiguo II-350
Yu, Shuiyuan I-117
Yu, Tong II-477
Yu, Yang II-274
Yuan, Kaiqi I-479
Yuan, Xiaojie I-29
Yue, Lin I-299, I-452
- Zeng, Qingtian I-415
Zhan, Xukuan I-461
Zhang, Aihua I-268
Zhang, Bohan II-403
Zhang, Deyuan II-283
Zhang, Fangtao II-221
- Zhang, Fangyuan II-266
Zhang, Haijiang II-274
Zhang, Hang I-377
Zhang, Haowen I-29
Zhang, Haoliang II-3
Zhang, Hualong I-125
Zhang, Jianpei II-403
Zhang, Junwei II-452
Zhang, Lei II-274
Zhang, Li I-226
Zhang, Liming II-48
Zhang, Ling II-144
Zhang, Liyuan I-276
Zhang, Qiang I-479
Zhang, Shuo I-148
Zhang, Weifeng I-213
Zhang, Xuefei II-465
Zhang, Yijia I-299
Zhang, Yonggang II-24
Zhang, Zili II-452
Zhao, Chenfei I-311
Zhao, Jiashi I-276
Zhao, Xin I-137
Zhao, Xuehua II-396
Zhao, Yu I-160
Zhou, Lihua I-403
Zhu, PeiDong I-433
Zhu, Xinhua I-16
Zhu, Zhenlong I-55
Zuo, Wanli I-299, I-452
Zuo, Xianglin II-361