

Xin Geng  
Byeong-Ho Kang (Eds.)

LNAI 11013

# PRICAI 2018: Trends in Artificial Intelligence

15th Pacific Rim  
International Conference on Artificial Intelligence  
Nanjing, China, August 28–31, 2018  
Proceedings, Part II

2  
Part II

 Springer

# Lecture Notes in Artificial Intelligence

11013

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>

Xin Geng · Byeong-Ho Kang (Eds.)

# PRICAI 2018: Trends in Artificial Intelligence

15th Pacific Rim  
International Conference on Artificial Intelligence  
Nanjing, China, August 28–31, 2018  
Proceedings, Part II

*Editors*  
Xin Geng  
Southeast University  
Nanjing  
China

Byeong-Ho Kang  
University of Tasmania  
Hobart, TAS  
Australia

ISSN 0302-9743                      ISSN 1611-3349 (electronic)  
Lecture Notes in Artificial Intelligence  
ISBN 978-3-319-97309-8              ISBN 978-3-319-97310-4 (eBook)  
<https://doi.org/10.1007/978-3-319-97310-4>

Library of Congress Control Number: 2018949307

LNCS Sublibrary: SL7 – Artificial Intelligence

© Springer International Publishing AG, part of Springer Nature 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

This volume contains the papers presented at the 15th Pacific Rim International Conference on Artificial Intelligence (PRICAI 2018) held during August 28–31, 2018, in Nanjing, China. PRICAI is a biennial conference inaugurated in Tokyo in 1990. It provides a common forum for researchers and practitioners in various branches of artificial intelligence (AI) to exchange new ideas and share experience and expertise. Over the past 28 years, the conference has grown, both in participation and scope, to be a premier international AI event for all major Pacific Rim nations as well as countries from further afield. This year, PRICAI 2018 featured two special tracks in addition to the main track, “Reinforcement Learning” and “Smart Modelling and Simulation,” both of which accentuated emerging hot topics in recent years of AI research.

This year, we received 382 high-quality submissions from 24 countries to both the main and special tracks. The submission number set a record over the last ten years (six PRICAIs in a row), reflecting the growing boom of artificial intelligence all over the world. The paper selection process was very competitive. From these submissions, 82 (21%) were accepted as regular papers, with a further 58 (15%) accepted as short papers. Each submitted paper was considered by the Program Committee (PC) members and external reviewers, and evaluated against criteria such as relevance, significance, technical soundness, novelty, and clarity. Every paper received at least two reviews, in most cases three, and in some cases up to five. Finally, the program co-chairs read the reviews, the original papers, and called for additional reviews if necessary to make final decisions. The entire review team (PC members, external reviewers, and co-chairs) expended tremendous effort to ensure fairness and consistency in the paper selection process.

The technical program consisted of workshops and tutorials and the three-day main conference program. There were four tutorials and six workshops covering thriving and important topics in artificial intelligence. The workshops included the Pacific Rim Knowledge Acquisition Workshop (PKAW), co-chaired by Kenichi Yoshida (University of Tsukuba, Japan) and Maria R. Lee (Shih Chien University, Taiwan, China), which has long enjoyed a successful co-location with PRICAI. All regular papers were orally presented over the three days in the topical program sessions and special sessions. The authors of short papers presented their results during the poster sessions, and were also offered the opportunity to give shortened talks to introduce their work. It was our great honor to have three outstanding keynote/invited speakers, whose contributions have pushed boundaries of artificial intelligence across various aspects: Professor Stephen Muggleton (Imperial College London, UK), Professor Qiang Yang (Hong Kong University of Science and Technology, China), and Dr. Kun Zhang (Carnegie Mellon University, USA). We are grateful to them for sharing their insights on their latest research with us.

The success of PRICAI 2018 would not have been possible without the effort and support of numerous people from all over the world. First of all, we would like to thank

the Program Committee members and external reviewers for their engagements in providing rigorous and timely reviews. It was because of them that the quality of the papers in this volume is maintained at a high level. We wish to express our gratitude to the general co-chairs, Zhi-Hua Zhou (Nanjing University, China) and Geoff Webb (Monash University, Australia) for their continued support and guidance. We are also thankful to the workshop co-chairs, Yang Yu (Nanjing University, China) and Tsuyoshi Murata (Tokyo Institute of Technology, Japan), the tutorial co-chairs, Mengjie Zhang (Victoria University of Wellington, New Zealand) and Tru Hoang Cao (Ho Chi Minh City University of Technology, Vietnam), the publication chair, Min-Ling Zhang (Southeast University, China), the sponsorship chair, Xiang Bai (Huazhong University of Science and Technology, China), the local organizing co-chairs, Feng Xu (Hohai University, China) and Deyu Zhou (Southeast University, China), and the publicity co-chairs, Chuan-Kang Ting (National Chung Cheng University, Taiwan, China) and Sheng-Jun Huang (Nanjing University of Aeronautics and Astronautics, China).

We gratefully acknowledge the support of the organizing institutions Southeast University, Jiangsu Association of Artificial Intelligence, Nanjing University, Nanjing University of Aeronautics and Astronautics, and Hohai University, as well as the financial support from Nanjing Future Sci-Tech City, Alibaba Group, Baidu Inc., Huatai Securities Co., Ltd., Huawei Technologies Co., Ltd., iHome Technologies Co., Ltd., Key Laboratory of IntelliSense Technology, CETC, Springer Publishing, and Jiangsu Zhitu Education Technology Co., Ltd. Special thanks to EasyChair, whose paper submission platform we used to organize reviews and collate the files for these proceedings. We are also grateful to Alfred Hofmann and Anna Kramer from Springer for their assistance in publishing the PRICAI 2018 proceedings as a volume in its *Lecture Notes in Artificial Intelligence* series.

Last but not least, we also want to thank all authors and all conference participants for their contribution and support. We hope all the participants took this valuable opportunity to share and exchange their ideas and thoughts with one another and enjoyed their time at PRICAI 2018.

August 2018

Xin Geng  
Byeong-Ho Kang

# Organization

## Steering Committee

Tru Hoang Cao	Ho Chi Minh City University of Technology, Vietnam
Aditya Ghose	University of Wollongong, Australia
Byeong-Ho Kang	University of Tasmania, Australia
Dickson Lukose	GCS Agile Pty. Ltd., Australia
Hideyuki Nakashima (Chair)	Future University Hakodate, Japan
Seong-Bae Park	Kyungpook National University, South Korea
Duc Nghia Pham	Griffith University, Australia
Abdul Sattar (Treasurer)	Griffith University, Australia
Ito Takayuki	Nagoya Institute of Technology, Japan
Thanaruk Theeramunkong	Sirindhorn International Institute of Technology, Thailand
Toby Walsh	NICTA, Australia
Zhi-Hua Zhou (Co-chair)	Nanjing University, China

## Organizing Committee

### General Co-chairs

Zhi-Hua Zhou	Nanjing University, China
Geoff Webb	Monash University, Australia

### Program Co-chairs

Xin Geng	Southeast University, China
Byeong-Ho Kang	University of Tasmania, Australia

### Workshop Co-chairs

Yang Yu	Nanjing University, China
Tsuyoshi Murata	Tokyo Institute of Technology, Japan

### Tutorial Co-chairs

Mengjie Zhang	Victoria University of Wellington, New Zealand
Tru Hoang Cao	Ho Chi Minh City University of Technology, Vietnam

### Publication Chair

Min-Ling Zhang	Southeast University, China
----------------	-----------------------------





Vlad Estivill-Castro	Griffith University, Australia
Christian Freksa	University of Bremen, Germany
Qiming Fu	Suzhou University of Science and Technology, China
Katsuhide Fujita	Tokyo University of Agriculture and Technology, Japan
Naoki Fukuta	Shizuoka University, Japan
Dragan Gamberger	Rudjer Boskovic Institute, Croatia
Wei Gao	Nanjing University, China
Xiaoying Gao	Victoria University of Wellington, New Zealand
Yang Gao	Nanjing University, China
Saurabh Kumar Garg	University of Tasmania, Australia
Xin Geng	Southeast University, China
Michael Granitzer	University of Passau, Germany
Fikret Gurgen	Bosphorus University, Turkey
Peter Haddawy	Mahidol University, Thailand
Bing Han	Xidian University, China
Soyeon Han	University of Tasmania, Australia
Choochart Haruechaiyasak	National Electronics and Computer Technology Center, Thailand
Kiyota Hashimoto	Prince of Songkla University, Thailand
Tessai Hayama	Nagaoka University of Technology, Japan
David Herbert	University of Tasmania, Australia
Chenping Hou	National University of Defense Technology, China
Juhua Hu	Simon Fraser University, Canada
Biwei Huang	Carnegie Mellon University, USA
Di Huang	Beihang University, China
Ko-Wei Huang	National Kaohsiung University of Science and Technology, Taiwan, China
Sheng-Jun Huang	Nanjing University of Aeronautics and Astronautics, China
Van Nam Huynh	JAIST, Japan
Masashi Inoue	Tohoku Institute of Technology, Japan
Sanjay Jain	National University of Singapore, Singapore
Jianmin Ji	University of Science and Technology of China, China
Binbin Jia	Southeast University, China
Liangxiao Jiang	China University of Geosciences, China
Mingmin Jiang	Southeast University, China
Yichuan Jiang	Southeast University, China
Yuan Jiang	Nanjing University, China
Hideaki Kanai	Japan Advanced Institute of Science and Technology, Japan
Ryo Kanamori	Nagoya University, Japan
Byeong-Ho Kang	University of Tasmania, Australia
Alfred Krzywicki	The University of New South Wales, Australia
Satoshi Kurihara	The University of Electro-Communications, Japan
Young-Bin Kwon	Chung-Ang University, South Korea

Weng Kin Lai	TARC, Malaysia
Ho-Pun Lam	CSIRO, Australia
Wee Sun Lee	National University of Singapore, Singapore
Roberto Legaspi	Research Organization of Information and Systems, The Institute of Statistical Mathematics, Japan
Gang Li	Deakin University, Australia
Guangliang Li	University of Amsterdam, The Netherlands
Li Li	Southwest University, China
Ming Li	Nanjing University, China
Nan Li	Alibaba Group, China
Tianrui Li	Southwest Jiaotong University, China
Wu-Jun Li	Nanjing University, China
Yu-Feng Li	Nanjing University, China
Beishui Liao	Zhejiang University, China
Miaogen Ling	Southeast University, China
Jiamou Liu	The University of Auckland, New Zealand
Liping Liu	Columbia University, USA
Mingxia Liu	The University of North Carolina at Chapel Hill, USA
Qing Liu	CSIRO, Australia
Ping Luo	Institute of Computing Technology, CAS; University of Chinese Academy of Sciences, China
Xudong Luo	Guangxi Normal University, China
Jiaqi Lv	Southeast University, China
Michael Maher	Reasoning Research Institute, Canberra, Australia
Xinjun Mao	National University of Defense Technology, China
Eric Martin	The University of New South Wales, Australia
Sanparith Marukatat	NECTEC, Thailand
James Montgomery	University of Tasmania, Australia
Koichi Moriyama	Nagoya Institute of Technology, Japan
Muhammad Marwan Muhammad Fuad	Technical University of Denmark, Denmark
Ekawit Nantajeewarawat	Thammasat University, Thailand
M. A. Hakim Newton	IIS, Griffith University, Australia
Su Nguyen	Victoria University of Wellington, New Zealand
Shahrul Azman Noah	Universiti Kebangsaan Malaysia, Malaysia
Masayuki Numao	Osaka University, Japan
Kouzou Ohara	Aoyama Gakuin University, Japan
Hayato Ohwada	Tokyo University of Science, Japan
Noriko Otani	Tokyo City University, Japan
Takanobu Otsuka	Nagoya Institute of Technology, Japan
Maurice Pagnucco	The University of New South Wales, Australia
Hye-Young Paik	The University of New South Wales, Australia
Jantima Polpinij	Maharakham University, Thailand
Chao Qian	University of Science and Technology of China, China
Yuhua Qian	Shanxi University, China
Joel Quinqueton	LIRMM, France

Ali Raza	University of Tasmania, Australia
Fenghui Ren	University of Wollongong, Australia
Yi Ren	Southeast University, China
Deborah Richards	Macquarie University, Australia
Kazumi Saito	University of Shizuoka, Japan
Chiaki Sakama	Wakayama University, Japan
Nicolas Schwind	Tokyo Institute of Technology, Japan
Rolf Schwitter	Macquarie University, Australia
Nazha Selmaoui-Folcher	University of New Caledonia, New Caledonia
Zhiqi Shen	Nanyang Technological University, Singapore
Chuan Shi	Beijing University of Posts and Telecommunications, China
Zhenwei Shi	Beihang University, China
Soo-Yong Shin	Kyung Hee University, South Korea
Shun Shiramatsu	Nagoya Institute of Technology, Japan
Yanfeng Shu	CSIRO, Australia
Waralak V. Siricharoen	Silpakorn University, Thailand
Tony Smith	University of Waikato, New Zealand
Chattrakul Sombattheera	Mahasarakham University, Thailand
Safeeullah Soomro	AMA International University, Bahrain
Markus Stumptner	University of South Australia, Australia
Hang Su	Tsinghua University, China
Xing Su	Beijing University of Technology, China
Merlin Teodosia Suarez	Center for Empathic Human-Computer Interactions, Philippines
Shiliang Sun	East China Normal University, China
Yanan Sun	Sichuan University, China
Boontawee Suntisrivaraporn	dtac, Thailand
Thepchai Supnithi	NECTEC, Thailand
Chang Wei Tan	Monash University, Australia
David Taniar	Monash University, Australia
Qing Tian	Nanjing University of Information Science and Technology, China
Binh Tran	Victoria University of Wellington, New Zealand
Shikui Tu	Shanghai Jiao Tong University, China
Miroslav Velez	Aries Design Automation, USA
Toby Walsh	The University of New South Wales, Australia
Dayong Wang	Nanyang Technological University, Singapore
Di Wang	Nanyang Technological University, Singapore
Jing Wang	Southeast University, China
Ke Wang	Southeast University, China
Kewen Wang	Griffith University, Australia
Qi Wang	Northwestern Polytechnical University, China
Rui Wang	Southeast University, China
Wei Wang	Nanjing University, China

Zhe Wang	East China University of Science and Technology, China
Yu-Wei Wen	National Chung Cheng University, Taiwan, China
Paul Weng	UM-SJTU Joint Institute, China
Yang Wenli	University of Tasmania, Australia
Wayne Wobcke	The University of New South Wales, Australia
Feng Wu	University of Science and Technology of China, China
Jiansheng Wu	Nanjing University of Posts and Telecommunications, China
Chang Xu	The University of Sydney, Australia
Guandong Xu	University of Technology Sydney, Australia
Ming Xu	Xi'an Jiaotong-Liverpool University, China
Ning Xu	Southeast University, China
Shuxiang Xu	University of Tasmania, Australia
Xin-Shun Xu	Shandong University, China
Hui Xue	Southeast University, China
Kong Yan	Nanjing University of Information, Science and Technology, China
Bo Yang	Jilin University, China
Meimei Yang	Southeast University, China
Ming Yang	Nanjing Normal University, China
Wanqi Yang	Nanjing Normal University, China
Roland Yap	National University of Singapore, Singapore
Dayong Ye	University of Wollongong, Australia
Kenichi Yoshida	University of Tsukuba, Japan
Chao Yu	University of Wollongong, Australia
Guoxian Yu	Southwest University, China
Han Yu	Nanyang Technological University, Singapore
Yang Yu	Nanjing University, China
Nanqi Yuan	University of Tasmania, Australia
Takaya Yuizono	Japan Advanced Institute of Science and Technology, Japan
Yifeng Zeng	Teesside University, UK
De-Chuan Zhan	Nanjing University, China
Chengqi Zhang	University of Technology Sydney, Australia
Daoqiang Zhang	Nanjing University of Aeronautics and Astronautics, China
Du Zhang	California State University, USA
Junping Zhang	Fudan University, China
Min-Ling Zhang	Southeast University, China
Minjie Zhang	University of Wollongong, Australia
Qieshi Zhang	Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, China
Shichao Zhang	Guangxi Normal University, China
Wen Zhang	Beijing University of Chemical Technology, China

Yu Zhang	The Hong Kong University of Science and Technology, China
Zhao Zhang	Soochow University, China
Zhaoxiang Zhang	Institute of Automation, Chinese Academy of Sciences, China
Zongzhang Zhang	Soochow University, China
Dengji Zhao	University of Southampton, UK
Li Zhao	Microsoft, China
Xiang Zhao	National University of Defense Technology, China
Yanchang Zhao	CSIRO, Australia
Shuigeng Zhou	Fudan University, China
Zhi-Hua Zhou	Nanjing University, China
Xiaofeng Zhu	Guangxi Normal University, China
Xingquan Zhu	Florida Atlantic University, USA
Fuzhen Zhuang	Institute of Computing Technology, Chinese Academy of Sciences, China
Quan Zou	Tianjin University, China

### **Additional Reviewers**

Ahmed Loai Ali	Chuanxing Geng	Xinyan Liang
Komate Amphawan	Mingming Gong	Jiahao Lin
Mohammad Dawud Ansari	Alireza Goudarzi	Aiden Liu
Molood Barati	Ryan Green	Chong Liu
Ying Bi	Qian Guo	Mengyu Liu
Sebastian Binnewies	Yuze Guo	Mingxia Liu
Alan Blair	Anasthasia Agnes Haryanto	Shaowu Liu
Weiling Cai	Shaojie He	Songtao Liu
Xinyuan Chen	Paramate Horkeaw	Yong Liu
Zixuan Chen	Yi-Qi Hu	Alex Long
Honghong Cheng	Kai Huang	Guoshuai Ma
Zhanzhan Cheng	Paul Salvador Inventado	Huifang Ma
Zhi Cheng	Saichon Jaiyen	Zhongchen Ma
Yao-Xiang Ding	Chong Jiang	Arturo Magana-Mora
Duy Tai Dinh	Johannes Jurgovsky	Wolfgang Mayer
Shaokang Dong	Abdul Karim	Koichi Moriyama
Yinpeng Dong	Bojana Kodric	Guodong Mu
Alan Downes	Longteng Kong	Mohsin Munir
Suhendry Effendy	Andre Kostenko	Courtney Ngo
Sara Elkarawi	Bin Li	Hung Ba Nguyen
Ji Feng	Chaoqun Li	Lei Niu
Zeyu Feng	Feijiang Li	Sebastian Palacio
Anna Förster	Weihua Li	Jianhui Pang
Longwen Gao	Yuyu Li	Ming Pang
		Manna Philip

Harun Pirim  
 Chen Qiu  
 Muhammad Imran Razzak  
 Adrien Rougny  
 Seung Ryu  
 Zafar Saeed  
 Fatemeh Salehi Rizi  
 Jörg Schlötterer  
 Matt Selway  
 Syed Wajid Ali Shah  
 Yang Shangdong  
 Xiang-Rong Sheng  
 John Shepherd  
 Leslie Sikos  
 Thanongchai Siriapisith  
 Fengyi Song  
 Kexiu Song  
 Sirawit Sopchoke  
 Changzhi Sun  
 Danyang Sun  
 Jia Sun  
 Jiahua Tang  
 Yanni Tang  
 Amit Thombre  
 Yanling Tian

Jannai Tokotoko  
 Tina Vajsbaher  
 Tina Vajsbaher-Kelc  
 Jasper van de Ven  
 Narumol Vannaprathip  
 Maria Vasardani  
 Nhi N. Y. Vo  
 Chaoyue Wang  
 Dengbao Wang  
 Hui Wang  
 Jieting Wang  
 Lu Wang  
 Xueping Wang  
 Yishen Wang  
 Yong Wang  
 Yuchen Wang  
 Yulong Wang  
 Zhiwei Wang  
 Fan Wu  
 Maonian Wu  
 Mingda Wu  
 Shiqing Wu  
 Song Wu  
 Xi-Zhu Wu  
 Yi-Feng Wu

Wei Xia  
 Peng Xiao  
 Zhang Xiaoyu  
 Qingsong Xie  
 Zhihao Xing  
 Junping Xu  
 Ziqi Yan  
 Fang Yang  
 Haote Yang  
 Yang Yang  
 Yi Yang  
 Haochao Ying  
 Vithya Yogarajan  
 Shan You  
 Liangjun Yu  
 Sicong Zang  
 Chenyu Zhang  
 Jihang Zhang  
 Weitong Zhang  
 Xiao Zhang  
 Xiaowei Zhao  
 Yao Zhou  
 Zili Zhou  
 Yunkai Zhuang  
 Zhiqiang Zhuang

## Organized by



Sponsored by

南京未来科技城





## Contents – Part II

An Improved Artificial Immune System Model for Link Prediction . . . . .	1
<i>Mengmeng Wang, Jianjun Ge, De Zhang, and Feng Zhang</i>	
Anchored Projection Based Capped $l_{2,1}$ -Norm Regression for Super-Resolution . . . . .	10
<i>Xiaotian Ma, Mingbo Zhao, Zhao Zhang, Jicong Fan, and Choujun Zhan</i>	
Query Expansion Based on Semantic Related Network . . . . .	19
<i>Limin Guo, Xing Su, Ling Zhang, Guangyan Huang, Xu Gao, and Zhiming Ding</i>	
Improving the Stability for Spiking Neural Networks Using Anti-noise Learning Rule. . . . .	29
<i>Yuling Luo, Qiang Fu, Junxiu Liu, Yongchuang Huang, Xuemei Ding, and Yi Cao</i>	
An Improved Convolutional Neural Network Model with Adversarial Net for Multi-label Image Classification . . . . .	38
<i>Tao Zhou, Zhixin Li, Canlong Zhang, and Lan Lin</i>	
Integrating Multiscale Contrast Regions for Saliency Detection. . . . .	47
<i>Taizhe Tan, Qunsheng Zeng, and Kangxi Xuan</i>	
Automatic Conditional Generation of Personalized Social Media Short Texts. . . . .	56
<i>Ziwen Wang, Jie Wang, Haiqian Gu, Fei Su, and Bojin Zhuang</i>	
Deep Multi-modal Learning with Cascade Consensus . . . . .	64
<i>Yang Yang, Yi-Feng Wu, De-Chuan Zhan, and Yuan Jiang</i>	
Driving the Narrative Flow of an Interactive Storytelling System for Case Studies . . . . .	73
<i>Stanley Yu Galan, Michael Joshua Ramos, Aakov Dy, Yusin Kim, and Ethel Ong</i>	
Pum-Riang Thai Silk Pattern Classification Using Texture Analysis . . . . .	82
<i>Kwankamon Dittakan and Nawanol Theera-Ampornpunt</i>	
Fuzzy Rough Based Feature Selection by Using Random Sampling. . . . .	91
<i>Wang Zhenlei, Zhao Suyun, Liu Yangming, Chen Hong, Li Cuiping, and Sun Xiran</i>	

Segmenting Sound Waves to Support Phonocardiogram Analysis: The PCGseg Approach . . . . .	100
<i>Hajar Alhijailan, Frans Coenen, Jo Dukes-McEwan, and Jeyarajan Thiyaalingam</i>	
A Lazy One-Dependence Classification Algorithm Based on Selective Patterns . . . . .	113
<i>Zhuoya Ju, Zhihai Wang, and Shiqiang Wang</i>	
A Client-Assisted Approach Based on User Collaboration for Indoor Positioning . . . . .	121
<i>Yiyi Zhang, Jun Tang, Michael Elimu, and Naizheng Bian</i>	
Achieving Multiagent Coordination Through CALA-rFMQ Learning in Continuous Action Space . . . . .	132
<i>Wanshu Liu, Chengwei Zhang, Tianpei Yang, Jianye Hao, Xiaohong Li, and Zhijie Bao</i>	
Environmental Reconstruction for Autonomous Vehicle Based on Image Feature Matching Constraint and Score . . . . .	140
<i>Fangchao Hu, Ling Bai, Yinguo Li, and Zhen Tian</i>	
An Improved Particle Filter Target Tracking Algorithm Based on Color Histogram and Convolutional Network . . . . .	149
<i>Shasha Gao, Liang Zhou, and Qiang Xie</i>	
Mini-Batch Variational Inference for Time-Aware Topic Modeling . . . . .	156
<i>Tomonari Masada and Atsuhiko Takasu</i>	
Using Differential Evolution to Estimate Labeler Quality for Crowdsourcing . . . . .	165
<i>Chen Qiu, Liangxiao Jiang, and Zhihua Cai</i>	
A Search Optimization Method for Rule Learning in Board Games . . . . .	174
<i>Hui Wang, Yanni Tang, Jiamou Liu, and Wu Chen</i>	
Image Segmentation Based on MRF Combining with Deep Learning Shape Priors . . . . .	182
<i>Yan Wang and Xili Wang</i>	
An Automated Matrix Profile for Mining Consecutive Repeats in Time Series . . . . .	192
<i>Mahtab Mirmomeni, Yousef Kowsar, Lars Kulik, and James Bailey</i>	
High-Resolution Depth Refinement by Photometric and Multi-shading Constraints. . . . .	201
<i>Yujun Zhang, Qian Zhang, and Wei Feng</i>	

Weakly-Supervised Object Localization by Cutting Background with Deep Reinforcement Learning . . . . .	210
<i>Wu Zheng and Zhaoxiang Zhang</i>	
Nature-Inspired Computational Model for Solving Bi-objective Traveling Salesman Problems . . . . .	219
<i>Xuejiao Chen, Zhengpeng Chen, Yingchu Xin, Xianghua Li, and Chao Gao</i>	
Differential Evolution-Based Weighted Majority Voting for Crowdsourcing . . . . .	228
<i>Hao Zhang, Liangxiao Jiang, and Wenqiang Xu</i>	
Scalable Machine Learning Techniques for Highly Imbalanced Credit Card Fraud Detection: A Comparative Study. . . . .	237
<i>Rafiq Ahmed Mohammed, Kok-Wai Wong, Mohd Fairuz Shiratuddin, and Xuequn Wang</i>	
View Decomposition and Adversarial for Semantic Segmentation . . . . .	247
<i>He Guan and Zhaoxiang Zhang</i>	
Efficient Bayesian Optimisation Using Derivative Meta-model . . . . .	256
<i>Ang Yang, Cheng Li, Santu Rana, Sunil Gupta, and Svetha Venkatesh</i>	
Prior Knowledge Guided Gene-Disease Associations Prediction: An Enhanced Inductive Matrix Completion Approach . . . . .	265
<i>Lei Chen, Jianyu Pu, Ziwen Yang, and Xingguo Chen</i>	
Text Classification with Enriched Word Features . . . . .	274
<i>Jingda Xu, Cheng Zhang, Peng Zhang, and Dawei Song</i>	
Attention-Based Linguistically Constraints Network for Aspect-Level Sentiment. . . . .	282
<i>Jinyu Lu and Yuexian Hou</i>	
Personalized POIs Travel Route Recommendation System Based on Tourism Big Data . . . . .	290
<i>Chenzhong Bin, Tianlong Gu, Yanpeng Sun, Liang Chang, Wenping Sun, and Lei Sun</i>	
Analysing TV Audience Engagement via Twitter: Incremental Segment-Level Opinion Mining of Second Screen Tweets . . . . .	300
<i>Gavin Katz, Bradford Heap, Wayne Wobcke, Michael Bain, and Sandeepa Kannangara</i>	

Absolute Orientation and Localization Estimation from an Omnidirectional Image. . . . . 309  
*Ruyu Liu, Jianhua Zhang, Kejie Yin, Zhiyin Pan, Ruihao Lin, and Shengyong Chen*

An Adaptive Clustering Algorithm by Finding Density Peaks. . . . . 317  
*Juanying Xie and Weiliang Jiang*

Statutes Recommendation Using Classification and Co-occurrence Between Statutes. . . . . 326  
*Yi Feng, Jidong Ge, Chuanyi Li, Li Kong, Feifei Zhang, and Bin Luo*

Robust and Real-Time Face Swapping Based on Face Segmentation and CANDIDE-3 . . . . . 335  
*Haosen Wang, Dongliang Xie, and Lu Wei*

Determining the Applicability of Advice for Efficient Multi-Agent Reinforcement Learning. . . . . 343  
*Yuchen Wang, Fenghui Ren, and Minjie Zhang*

Multi-object Detection Based on Deep Learning in Real Classrooms. . . . . 352  
*Benchi Shao, Fei Jiang, and Ruimin Shen*

Deep CRF-Graph Learning for Semantic Image Segmentation . . . . . 360  
*Fuguang Ding, Zhenhua Wang, Dongyan Guo, Shengyong Chen, Jianhua Zhang, and Zhanpeng Shao*

Unrest News Amount Prediction with Context-Aware Attention LSTM . . . . . 369  
*Xiuling Wang, Hao Chen, Zhoujun Li, and Zhonghua Zhao*

Image Captioning with Relational Knowledge. . . . . 378  
*Huan Yang, Dandan Song, and Lejian Liao*

An Elite Group Guided Artificial Bee Colony Algorithm with a Modified Neighborhood Search. . . . . 387  
*Jiaxin Lu, Xinyu Zhou, Yong Ma, and Mingwen Wang*

Exploiting Spatiotemporal Features to Infer Friendship in Location-Based Social Networks . . . . . 395  
*Cheng He, Chao Peng, Na Li, Xiang Chen, and Lanying Guo*

A Subsequent Speaker Selection Method for Online Discussions Based on the Multi-armed Bandit Algorithm. . . . . 404  
*Mio Kurii and Katsuhide Fujita*

An Entropy-Based Class Assignment Detection Approach for RDF Data . . . . . 412  
*Molood Barati, Quan Bai, and Qing Liu*

Weighted Double Deep Multiagent Reinforcement Learning in Stochastic Cooperative Environments . . . . . <i>Yan Zheng, Zhaopeng Meng, Jianye Hao, and Zongzhang Zhang</i>	421
Automatically Classifying Chinese Judgment Documents Using Character-Level Convolutional Neural Networks . . . . . <i>Xiaosong Zhou, Chuanyi Li, Jidong Ge, Zhongjin Li, Xiaoyu Zhou, and Bin Luo</i>	430
RC-CNN: Reverse Connected Convolutional Neural Network for Accurate Player Detection . . . . . <i>Lijing Zhang, Yao Lu, Ge Song, and Hanfeng Zheng</i>	438
Uncertainty Estimation for Strong-Noise Data. . . . . <i>Bin Shen and Binheng Song</i>	447
Reciprocal Ranking: A Hybrid Ranking Algorithm for Reciprocal Recommendation . . . . . <i>Yuanhang Qu, Hongzhi Liu, Yingpeng Du, and Zhonghai Wu</i>	455
Robust Low-Rank Recovery with a Distance-Measure Structure for Face Recognition . . . . . <i>Zhe Chen, Xiao-Jun Wu, He-Feng Yin, and Josef Kittler</i>	464
A Surface Defect Detection Method Based on Positive Samples . . . . . <i>Zhixuan Zhao, Bo Li, Rong Dong, and Peng Zhao</i>	473
Joint Multi-field Siamese Recurrent Neural Network for Entity Resolution . . . . . <i>Yang Lv, Lei Qi, Jing Huo, Hao Wang, and Yang Gao</i>	482
Using Machine Learning for Determining Network Robustness of Multi-Agent Systems Under Attacks . . . . . <i>Guang Wang, Ming Xu, Yiming Wu, Ning Zheng, Jian Xu, and Tong Qiao</i>	491
Collective Hyper-heuristics for Self-assembling Robot Behaviours. . . . . <i>Shuang Yu, Andy Song, and Aldeida Aleti</i>	499
Matrix Factorization for Identifying Noisy Labels of Multi-label Instances . . . . . <i>Xia Chen, Guoxian Yu, Carlotta Domeniconi, Jun Wang, and Zili Zhang</i>	508
<b>Author Index . . . . .</b>	519

# Contents – Part I

HAVAE: Learning Prosodic-Enhanced Representations of Rap Lyrics . . . . .	1
<i>Hongru Liang, Qian Li, Haozheng Wang, Hang Li, Jun Wang, Zhe Sun, Jin-Mao Wei, and Zhenglu Yang</i>	
DKE-RLS: A Manifold Reconstruction Algorithm in Label Spaces with Double Kernel Embedding-Regularized Least Square . . . . .	16
<i>Chao Tan and Genlin Ji</i>	
Learning Relations from Social Tagging Data . . . . .	29
<i>Hang Dong, Wei Wang, and Frans Coenen</i>	
Selecting Optimal Source for Transfer Learning in Bayesian Optimisation . . .	42
<i>Anil Ramachandran, Sunil Gupta, Santu Rana, and Svetha Venkatesh</i>	
Fast Spatially-Regularized Correlation Filters for Visual Object Tracking . . .	57
<i>Pengyu Zhang, Qing Guo, and Wei Feng</i>	
Similarity-Adaptive Latent Low-Rank Representation for Robust Data Representation . . . . .	71
<i>Lei Wang, Zhao Zhang, Sheng Li, Guangcan Liu, Chenping Hou, and Jie Qin</i>	
Adaptively Shaping Reinforcement Learning Agents via Human Reward . . .	85
<i>Chao Yu, Dongxu Wang, Tianpei Yang, Wenxuan Zhu, Yuchen Li, Hongwei Ge, and Jiankang Ren</i>	
Incomplete Multi-view Clustering via Structured Graph Learning . . . . .	98
<i>Jie Wu, Wenzhang Zhuge, Hong Tao, Chenping Hou, and Zhao Zhang</i>	
DeepRSD: A Deep Regression Method for Sequential Data . . . . .	113
<i>Xishun Wang, Minjie Zhang, and Fenghui Ren</i>	
Single Image Super-Resolution via Perceptual Loss Guided by Denoising Auto-Encoder . . . . .	126
<i>Zhong-Han Niu, Lu-Fei Liu, Kai-Jun Zhang, Jian-Feng Dong, Yu-Bin Yang, and Xiao-Jiao Mao</i>	
Context-Aware Phrase Representation for Statistical Machine Translation . . .	137
<i>Zhiwei Ruan, Jinsong Su, Deyi Xiong, and Rongrong Ji</i>	

Collaborating Aesthetic Change and Heterogeneous Information into Recommender Systems . . . . .	150
<i>Zongze Jin, Yun Zhang, Weimin Mu, Weiping Wang, and Hai Jin</i>	
Latent Subspace Representation for Multiclass Classification . . . . .	163
<i>Jing Hu, Changqing Zhang, Xiao Wang, Pengfei Zhu, Zheng Wang, and Qinghua Hu</i>	
Low-Rank Graph Regularized Sparse Coding . . . . .	177
<i>Yupei Zhang, Shuhui Liu, Xuequn Shang, and Ming Xiang</i>	
Decentralized Multiagent Reinforcement Learning for Efficient Robotic Control by Coordination Graphs . . . . .	191
<i>Chao Yu, Dongxu Wang, Jiankang Ren, Hongwei Ge, and Liang Sun</i>	
Construction of Microblog-Specific Chinese Sentiment Lexicon Based on Representation Learning. . . . .	204
<i>Li Kong, Chuanyi Li, Jidong Ge, Yufan Yang, Feifei Zhang, and Bin Luo</i>	
Phonologically Aware BiLSTM Model for Mongolian Phrase Break Prediction with Attention Mechanism . . . . .	217
<i>Rui Liu, FeiLong Bao, Guanglai Gao, Hui Zhang, and Yonghe Wang</i>	
Multi-label Crowdsourcing Learning with Incomplete Annotations . . . . .	232
<i>Shao-Yuan Li and Yuan Jiang</i>	
Multiple Kernel Fusion with HSIC Lasso. . . . .	246
<i>Tinghua Wang and Fulai Liu</i>	
Visualizing and Understanding Policy Networks of Computer Go . . . . .	256
<i>Yuanfeng Pang and Takeshi Ito</i>	
A Multi-objective Optimization Model for Determining the Optimal Standard Feasible Neighborhood of Intelligent Vehicles . . . . .	268
<i>Lei Huang, Ying Xu, and Hailiang Zhao</i>	
Efficient Detection of Critical Links to Maintain Performance of Network with Uncertain Connectivity . . . . .	282
<i>Kazumi Saito, Kouzou Ohara, Masahiro Kimura, and Hiroshi Motoda</i>	
Mixed Neighbourhood Local Search for Customer Order Scheduling Problem. . . . .	296
<i>Vahid Riahi, M. M. A. Polash, M. A. Hakim Newton, and Abdul Sattar</i>	

Graph Based Family Relationship Recognition from a Single Image . . . . .	310
<i>Chao Xia, Siyu Xia, Yuan Zhou, Le Zhang, and Ming Shao</i>	
ACGAIL: Imitation Learning About Multiple Intentions with Auxiliary Classifier GANs . . . . .	321
<i>Jiahao Lin and Zongzhang Zhang</i>	
Matching Attention Network for Domain Adaptation Optimized by Joint GANs and KL-MMD . . . . .	335
<i>Yuan-Zhu Gan, Hai-Qing Wang, Lu-Fei Liu, and Yu-Bin Yang</i>	
Attention Based Meta Path Fusion for Heterogeneous Information Network Embedding. . . . .	348
<i>Houye Ji, Chuan Shi, and Bai Wang</i>	
An Efficient Auction with Variable Reserve Prices for Ridesourcing . . . . .	361
<i>Chaoli Zhang, Fan Wu, and Xiaohui Bei</i>	
Matrix Entropy Driven Maximum Margin Feature Learning . . . . .	375
<i>Dong Zhang, Jinhui Tang, and Zechao Li</i>	
Spectral Image Visualization Using Generative Adversarial Networks . . . . .	388
<i>Siyu Chen, Danping Liao, and Yuntao Qian</i>	
Fusing Semantic Prior Based Deep Hashing Method for Fuzzy Image Retrieval . . . . .	402
<i>Xiaolong Gong, Linpeng Huang, and Fuwei Wang</i>	
Topic-Guided Automatic Human-Simulated Tweeting System . . . . .	416
<i>Zongyue Liu, Fuhai Chen, Jinsong Su, Chen Shen, and Rongrong Ji</i>	
Network Embedding Based on a Quasi-Local Similarity Measure . . . . .	429
<i>Xin Liu, Natthawut Kertkeidkachorn, Tsuyoshi Murata, Kyoung-Sook Kim, Julien Leblay, and Steven Lynden</i>	
Reinforcement Learning for Mobile Robot Obstacle Avoidance Under Dynamic Environments . . . . .	441
<i>Liwei Huang, Hong Qu, Mingsheng Fu, and Wu Deng</i>	
Subclass Maximum Margin Tree Error Correcting Output Codes. . . . .	454
<i>Fa Zheng and Hui Xue</i>	
A Multi-latent Semantics Representation Model for Mining Tourist Trajectory . . . . .	463
<i>Yanpeng Sun, Tianlong Gu, Chenzhong Bin, Liang Chang, Haili Kuang, Zhaowei Huang, and Lei Sun</i>	



Two-Stage Unsupervised Deep Hashing for Image Retrieval . . . . .	477
<i>Yuan-Zhu Gan, Hao Hu, and Yu-Bin Yang</i>	
A Fast Heuristic Path Computation Algorithm for the Batch Bandwidth Constrained Routing Problem in SDN . . . . .	490
<i>Dongjun Qian, Peng Yang, and Ke Tang</i>	
3SP-Net: Semantic Segmentation Network with Stereo Image Pairs for Urban Scene Parsing . . . . .	503
<i>Lingli Zhou and Haofeng Zhang</i>	
An Interactivity-Based Personalized Mutual Reinforcement Model for Microblog Topic Summarization . . . . .	518
<i>Lu Zhang, Liangjun Zang, Longtao Huang, Jizhong Han, and Songlin Hu</i>	
Multiple Visual Fields Cascaded Convolutional Neural Network for Breast Cancer Detection . . . . .	531
<i>Haomiao Ni, Hong Liu, Zichao Guo, Xiangdong Wang, Taijiao Jiang, Kuansong Wang, and Yueliang Qian</i>	
Multi-view Learning and Deep Learning for Microscopic Neuroblastoma Pathology Image Diagnosis . . . . .	545
<i>Yuhan Liu, Minzhi Yin, and Shiliang Sun</i>	
Low-Rank Matrix Recovery via Continuation-Based Approximate Low-Rank Minimization . . . . .	559
<i>Xiang Zhang, Yongqiang Gao, Long Lan, Xiaowei Guo, Xuhui Huang, and Zhigang Luo</i>	
Inertial Constrained Hierarchical Belief Propagation for Optical Flow . . . . .	574
<i>Zixing Zhang and Ying Wen</i>	
ParallelNet: A Depth-Guided Parallel Convolutional Network for Scene Segmentation . . . . .	588
<i>Shiyu Liu and Haofeng Zhang</i>	
Aircraft Detection in Remote Sensing Images Based on Background Filtering and Scale Prediction . . . . .	604
<i>Jing Gao, Haichang Li, Zhongxing Han, Siyu Wang, and Xiaohui Hu</i>	
Residual Convolutional Neural Networks with Global and Local Pathways for Classification of Focal Liver Lesions . . . . .	617
<i>Dong Liang, Lanfen Lin, Hongjie Hu, Qiaowei Zhang, Qingqing Chen, Yutaro Iwamoto, Xianhua Han, and Yen-Wei Chen</i>	

Intent Detection for Spoken Language Understanding Using a Deep Ensemble Model . . . . .	629
<i>Mauajama Firdaus, Shobhit Bhatnagar, Asif Ekbal, and Pushpak Bhattacharyya</i>	
Accurately Detecting Community with Large Attribute in Partial Networks . . . . .	643
<i>Wei Han, Guopeng Li, and Xinyu Zhang</i>	
Two-Step Multi-factor Attention Neural Network for Answer Selection . . . . .	658
<i>Pengqing Zhang, Yuexian Hou, Zhan Su, and Yi Su</i>	
Labeling Information Enhancement for Multi-label Learning with Low-Rank Subspace . . . . .	671
<i>An Tao, Ning Xu, and Xin Geng</i>	
Deep Coordinated Textual and Visual Network for Sentiment-Oriented Cross-Modal Retrieval . . . . .	684
<i>Jiamei Fu, Dongyu She, Xingxu Yao, Yuxiang Zhang, and Jufeng Yang</i>	
A New Context-Based Clustering Framework for Categorical Data . . . . .	697
<i>Thanh-Phu Nguyen, Duy-Tai Dinh, and Van-Nam Huynh</i>	
TypicFace: Dynamic Margin Cosine Loss for Deep Face Recognition . . . . .	710
<i>Lei Li, Heng Luo, Lei Zhang, Qing Xu, and Hao Ning</i>	
Semi-supervised Feature Selection Based on Logistic I-RELIEF for Multi-classification . . . . .	719
<i>Baige Tang and Li Zhang</i>	
Genetic Programming for Feature Selection and Feature Construction in Skin Cancer Image Classification . . . . .	732
<i>Qurrat Ul Ain, Bing Xue, Harith Al-Sahaf, and Mengjie Zhang</i>	
Unsupervised Stereo Matching with Occlusion-Aware Loss . . . . .	746
<i>Ningqi Luo, Chengxi Yang, Wenxiu Sun, and Binheng Song</i>	
Siamese Network Based Features Fusion for Adaptive Visual Tracking . . . . .	759
<i>Dongyan Guo, Weixuan Zhao, Ying Cui, Zhenhua Wang, Shengyong Chen, and Jian Zhang</i>	
ANNC: AUC-Based Feature Selection by Maximizing Nearest Neighbor Complementarity . . . . .	772
<i>Xuemeng Jiang, Jun Wang, Jinmao Wei, Jianhua Ruan, and Gang Yu</i>	

Prediction of Nash Bargaining Solution in Negotiation Dialogue . . . . .	786
<i>Kosui Iwasa and Katsuhide Fujita</i>	
Joint Residual Pyramid for Depth Map Super-Resolution . . . . .	797
<i>Yi Xiao, Xiang Cao, Yan Zheng, and Xianyi Zhu</i>	
Reading More Efficiently: Multi-sentence Summarization with a Dual Attention and Copy-Generator Network . . . . .	811
<i>Xi Zhang, Hua-ping Zhang, and Lei Zhao</i>	
Staged Generative Adversarial Networks with Adversarial-Boundary . . . . .	824
<i>Zhifan Li, Dandan Song, and Lejian Liao</i>	
Semi-supervised DenPeak Clustering with Pairwise Constraints. . . . .	837
<i>Yazhou Ren, Xiaohui Hu, Ke Shi, Guoxian Yu, Dezhong Yao, and Zenglin Xu</i>	
A Novel Convolutional Neural Network for Statutes Recommendation . . . . .	851
<i>Chuanyi Li, Jingjing Ye, Jidong Ge, Li Kong, Haiyang Hu, and Bin Luo</i>	
Towards Understanding User Requests in AI Bots. . . . .	864
<i>Oanh Thi Tran and Tho Chi Luong</i>	
A Deep Reinforced Training Method for Location-Based Image Captioning . . . . .	878
<i>Lei Zhao, Chunxia Zhang, Xi Zhang, Yating Hu, and Zhendong Niu</i>	
Graph Stream Mining Based Anomalous Event Analysis . . . . .	891
<i>Meng Yang, Lida Rashidi, Sutharshan Rajasegarar, and Christopher Leckie</i>	
Nonlinearized Relevance Propagation . . . . .	904
<i>Quexuan Zhang and Yukio Ohsawa</i>	
Online Personalized Next-Item Recommendation via Long Short Term Preference Learning . . . . .	915
<i>Yingpeng Du, Hongzhi Liu, Yuanhang Qu, and Zhonghai Wu</i>	
Enhancing Artificial Bee Colony Algorithm with Superior Information Learning . . . . .	928
<i>Xinyu Zhou, Yunan Liu, Mingwen Wang, and Jianyi Wan</i>	
Robust Factorization Machines for Credit Default Prediction . . . . .	941
<i>Weijian Ni, Tong Liu, Qingtian Zeng, Xianke Zhang, Hua Duan, and Nengfu Xie</i>	

Multi-label Active Learning with Conditional Bernoulli Mixtures . . . . . 954  
*Junyu Chen, Shiliang Sun, and Jing Zhao*

Investigating the Dynamic Decision Mechanisms of Users’ Relevance  
Judgment for Information Retrieval via Log Analysis . . . . . 968  
*Yi Su, Jingfei Li, Dawei Song, Pengqing Zhang, and Yazhou Zhang*

A Correlation-Aware ML-kNN Algorithm for Customer Value Modeling  
in Online Shopping . . . . . 980  
*Yuan Zhuang, Xiaolin Li, Yue Sun, and Xiangdong He*

Binary Collaborative Filtering Ensemble . . . . . 993  
*Yujia Zhang, Jun Wu, and Haishuai Wang*

Social Collaborative Filtering Ensemble . . . . . 1005  
*Honglei Zhang, Gangdu Liu, and Jun Wu*

High-Performance OCR on Packing Boxes in Industry Based  
on Deep Learning . . . . . 1018  
*Fei Chen, Bo Li, Rong Dong, and Peng Zhao*

An Implementation of Large-Scale Holonic Multi-agent Society Simulator  
and Agent Behavior Model . . . . . 1031  
*Takayuki Ito, Takanobu Otsuka, Teruyoshi Imaeda, and Rafik Hadfi*

Establishing Connections in a Social Network: Radial Versus Medial  
Centrality Indices . . . . . 1044  
*Yanni Tang, Jiamou Liu, Wu Chen, and Zhuoxing Zhang*

Gradient Hyperalignment for Multi-subject fMRI Data Alignment. . . . . 1058  
*Tonglin Xu, Muhammad Yousefnezhad, and Daoqiang Zhang*

Node Based Row-Filter Convolutional Neural Network for Brain Network  
Classification . . . . . 1069  
*Bingcheng Mao, Jiashuang Huang, and Daoqiang Zhang*

**Author Index** . . . . . 1081



# An Improved Artificial Immune System Model for Link Prediction

Mengmeng Wang<sup>(✉)</sup>, Jianjun Ge, De Zhang, and Feng Zhang

Information Science Academy, China Electronics Technology Group Corporation, Beijing, China

wangmm\_cetc\_isa@163.com, gejj\_cetc\_isa@163.com,  
zhangd\_cetc\_isa@163.com, zhangf\_cetc\_isa@163.com

**Abstract.** Currently, online social network has derived a series of hot research problems, such as link prediction. Many results in undirected and dynamic network have been achieved. Targeted at on-line microblogs, this paper first build user's dynamic emotional indices and network topological structure features based on time series of user's contents and network topological information. Then, we improve artificial immune system and deploy it to predict the existence and direction of link. Experiments on real-world dataset demonstrate the effectiveness of the proposed framework. Further experiments are conducted to understand the importance of temporal information in link prediction.

**Keywords:** Link prediction · Improved artificial immune system  
Time series · Emotional indices · Dynamic social network

## 1 Introduction

Nowadays, online social networks are growing and becoming denser [1]. The social connections of a given person may have very high variability [2]. Consequently, link prediction is attracting increasing interests among data mining and machine learning communities [3]. It has many applications, such as social recommendation [4], sentiment analysis [5], spam comments detection [6], and so forth. As stated, our work on predicting links is motivated by its broad application prospect.

In this paper, we propose an improved artificial immune system model for link prediction (denoted as IAISLP). The main contributions are summarized next.

- Improve traditional Salton metrics to achieve a better representation about adjacent degree between nodes in directed social network.
- Blend temporal information in link prediction algorithm through building dynamic relationship-based and emotional-based features based on time series of user's contents and network topological information.
- Improve artificial immune system with novel affinity measurement, multifarious affinity thresholds and normally distributed mutation operator in order to adapt to individual diversity, followed by employing it to predict the existence and direction of link.

The rest of paper is organized as follows: Sect. 2 describes the related work; Sect. 3 defines the method we propose; Details of the experimental results and dataset which is used in this study are given in Sect. 4. Finally conclusion appears in Sect. 5.

## 2 Related Work

Development of modeling theory of link prediction has been promoted since a variety of mathematical methods were utilized to establish link prediction model [7].

With the deepening of social network mining, some researchers paid more attention on additional information to improve performance of algorithms. In contrast to most works that used interaction data between users, which were private and thus, typically not available, Rozenshtein et al. also considered a collection of tight communities as input [2]. Besides, Wei et al. studied the problem of cross view link prediction on partially observable networks thoroughly, where the focus was to recommend nodes with only links to nodes with only attributes (or vice versa) [8].

With the availability of evolving social network data, recent studies considered temporal link prediction problem [9]. Zhang et al. raised latent friendship transitivity tree, in which growth process of a node's friends network was represented as spanning tree structure, besides root node, position of each node in the tree was decided by the time that established relations with root node [10].

In the current, there are also some researches on link direction predicting. Fire et al. proposed a direction-aware proximity method and utilized J48 decision tree, Bagging and Random Forest for link prediction [11]. Schall put forward a measure method of adjacent degree in view of ternary closed relationship to forecast link between nodes via graph model [12].

To sum up, link existence and direction prediction in social networks is in the stage of development, how to depict directionality of linkages, fuse multidimensional features reasonably and build model that could predict links efficiently can be very challenging jobs. To this end, we present an improved artificial immune system model which is appropriate for link prediction in dynamic and directed social network.

## 3 Improved Artificial Immune System Model for Link Prediction

### 3.1 Definition of Dynamic Link Prediction Features

**Profile-Based Features.** Users who are similar in their profile are more likely to establish link with each other. In this paper, user's profile features are mainly consisted of the number of contents that user post, bifollowers, followers and friends, gender, province and city, created time and verified type of user's account, where discrete values of gender, province, city and verified type of user's account are already represented with different numeric values in the original dataset.

**Dynamic Relationship-Base Features.** Traditional link prediction algorithm held the view that the more similar users' network topological structures were, the easier they became friends with each other. Consequently, in this work, we utilize Salton metrics to infer probability of link establishment between users. However, traditional Salton metrics have not defined direction of links and relationships would decay with time [13]. Thus, we consider network as a dynamic flow of time slices with different weights, and a series of improved topological metrics are denoted as relationship-based features in an independent time slices. Salton metrics of user  $u$  and  $v$  on the  $i$ -th time slice  $t_i$  as follows.

$$Sa(u, v, t_i) = \frac{|\Gamma^{in}(u, t_i) \cap \Gamma^{in}(v, t_i)| / \sqrt{|\Gamma^{in}(u, t_i)| \times |\Gamma^{in}(v, t_i)|}}{|\Gamma^{out}(u, t_i) \cap \Gamma^{out}(v, t_i)| / \sqrt{|\Gamma^{out}(u, t_i)| \times |\Gamma^{out}(v, t_i)|}} \quad (1)$$

Where  $\Gamma^{in}(u, t_i)$ ,  $\Gamma^{in}(v, t_i)$ ,  $\Gamma^{out}(u, t_i)$  and  $\Gamma^{out}(v, t_i)$  stand for in-link and out-link users set of  $u$  and  $v$  on  $t_i$  respectively, and in-link and out-link are defined by follower relationship,  $|\cdot|$  stand for the number of elements in a set. Thus, improved Salton metrics of  $u$  and  $v$  on the flow of time slices  $[0, t_n]$  is calculated as:

$$Sa^{[0, t_n]}(u, v) = \sum_{i=0}^n \alpha^{n-i} \times Sa(u, v, t_i) \quad (2)$$

where  $\alpha^{n-i}$  represents the weight of  $t_i$ .

**Dynamic Emotion-Based Feature.** Some works demonstrated that relationship social closeness information can be inferred from linguistic features [14]. Hence, we conduct sentiment analysis on user's contents with corpus of HowNet Knowledge, which includes 8945 words and phrases, is consists of positive and negative emotional words list files, positive and negative review words list files.

*Definition 1.* A user's emotional indices denotes emotional tendency expressed in his/her contents.

In the same way of dynamic relationship-based features, user  $u$ 's emotional indices on  $t_i$  is calculated as follows:

$$Em(u, t_i) = pn(u, t_i) / nn(u, t_i) \quad (3)$$

where  $Em(u, t_i)$  represents emotional indices of  $u$ 's content on  $t_i$ ,  $pn(u, t_i)$  and  $nn(u, t_i)$  represent the number of positive emotional words and negative emotional words  $u$  used on  $t_i$  which are included in HowNet Knowledge. Thus, emotional indices of user  $u$  and  $v$  on  $[0, t_n]$  is calculated as:

$$Eml^{[0, t_n]}(u, v) = \sum_{i=0}^n \alpha^{n-i} \times (Em(u, t_i) - Em(v, t_i)) \quad (4)$$

### 3.2 Algorithm for Link Prediction

From a computational perspective, artificial immune system, which recognizes nonself antigens (such as bacteria, viruses) via continual learning, can adapt to diversities of features well. Since links and users' attributes can usually be partially observable, so we build a framework based on an improved artificial immune system model for link prediction. Here, we classify link between user  $u$  and  $v$  in online social networks into three categories:

- Positive link, namely link exists and its direction is from user  $u$  to  $v$ ;
- Negative link, namely link exists and its direction is from user  $v$  to  $u$ ;
- Nonexistent link, namely there is no link between user  $u$  and  $v$ .

Since positive link, negative link and nonexistent link are recognized as nonself in our proposed framework, so in this paper, a link can be treated as an antigen in artificial immune system. The detailed descriptions of our method are shown as follows.

**Stage of Initializing Antigen and Antibody Set.** First, add all links into initial antigen link set  $AG$ . Then random select  $AB \subseteq AG$  as initial antibody link set ( $AB$  is not excluded from  $AG$ ), followed by initializing generation of these links in  $AB$  as 0 which will be introduced in detail later.

**Stage of Generating Memory Cells.** In this phase,  $G$  times of iterations are carried on. In each time of iteration, there are four steps.

Step 1: calculating affinity threshold of each antibody link  $ab$  in  $AB$ . If all antibody links employ a common threshold, each antibody link will only cover a certain space and cannot reach its maximum efficiency. To fix this, we set the initial threshold of each antibody link  $ab$  (denoted as  $\theta_{ab}$ ) as 0.5 and modify  $\theta_{ab}$  in each iteration as:

$$\theta_{ab} = \theta_{ab} + \delta \times \sum_{ag \in AG} (Aff(ab, ag) - \theta_{ab}) \quad (5)$$

where  $\delta$  denotes a monotonic decreasing learning rate,  $Aff(ab, ag)$  denotes improved affinity between  $ab$  and antigen link  $ag$  which is calculated as:

$$Aff(ab, ag) = 2^{-\lambda \times life_{ab}} / D_{JS}(V_{ab}, V_{ag}) \quad (6)$$

where  $\lambda \in [0, 1]$  represents a decay factor,  $life_{ab} \in \{0, 1, \dots, G\}$  represents generation of  $ab$ ,  $V_{ab}$  and  $V_{ag}$  represent  $ab$  and  $ag$ 's feature vector,  $D_{JS}(V_{ab}, V_{ag})$  represents  $V_{ab}$  and  $V_{ag}$ 's Jensen-Shannon divergence which is calculated as:

$$D_{JS}(V_{ab}, V_{ag}) = (D_{KL}(V_{ab} \| R) + D_{KL}(V_{ag} \| R)) / 2 \quad (7)$$

where  $R = 1/2(V_{ab} + V_{ag})$  denotes an average distribution of  $V_{ab}$  and  $V_{ag}$ ,  $D_{KL}(V_{ab} \| R)$  denotes Kullback-Leibler divergence between  $V_{ab}$  and  $R$ , which is calculated as follows:

$$D_{KL}(V_{ab} \| R) = \sum_i V_{ab}(i) \log \frac{V_{ab}(i)}{R(i)} \quad (8)$$



where  $V_{ab}(i)$  and  $R(i)$  denote the  $i$ -th attribute's value of  $V_{ab}$  and  $R$ .  $D_{KL}(V_{ag}||R)$  can be obtained by the same reason.

Step 2: streamline antibody link set  $AB$ . Calculate  $ab$ 's fitness as:

$$fit(ab) = cNum_{ab}/fNum_{ab} \quad (9)$$

where  $cNum_{ab}$  and  $fNum_{ab}$  represent the number of antigen links that  $ab$  classifies correctly and falsely respectively. Then delete any antibody link whose fitness is less than the predetermined threshold  $\varepsilon$  and add streamlined antibody link set  $AB$  into cloning and mutating candidate link set  $AB^*$ .

Step 3: cloning and mutating antibody links in  $AB^*$ . In this paper, an improved clonal selection algorithm is utilized to complete antibody links' clone and mutation.

First, mutate each attribute of each  $ab$  in  $AB^*$ . The range of  $ab$ 's  $i$ -th attribute is calculated as follows:

$$ran_{ab}(i) = (up_i - down_i) \times fit(ab) / \sum_{h=1}^{|AB^*|} fit(h) \quad (10)$$

where  $up_i$  and  $down_i$  denote upper limit value and lower limit value of the  $i$ -th attribute respectively. Adopting a well-proportioned mutation operator will make mutated values distribute evenly in their value scope. Consequently, a novel mutation operator is employed to make mutated values follow a normal distribution which will not only reduce the probability of having a too large or too small mutation range, but also find the optimal solution more quickly. The mutated antibody link  $ab^*$ 's  $i$ -th attribute is calculated as follows:

$$V_{ab^*}(i) = down_i + ran_{ab}(i) \times \rho \quad (11)$$

where  $\rho \in [0, 1]$  denotes a random value that follows a normal distribution.

Then clone each  $ab^*$ .  $ab^*$ 's number of clones is calculated as follows:

$$clNum(ab^*) = \left\lfloor fit(ab) \times |AG| \times \beta / \sum_{h=1}^{|AB^*|} fit(h) \right\rfloor \quad (12)$$

where  $\beta$  stands for a proportional factor of cloning.

Finally, add clonal antibodies into  $AB$  and set them as the next generation of  $ab$ .

Step 4: selecting mature antibody links. Calculate each antibody link's threshold and fitness in  $AB$ , if there is an antibody link whose fitness is lower than  $\varepsilon$ , then delete it from  $AB$ , otherwise, add it into memory cell link set  $MC$ .

**Stage of Predicting.** Firstly, calculate affinity between unknown link and each memory cell link in  $MC$  with (6). Secondly, identify unknown link's category according to the label of memory cell which has the highest affinity with unknown link.

## 4 Experimental Evaluation

In this section, we conduct experiments to assess the effectiveness of the proposed framework IAISLP. Through the experiments, we aim to answer the following two questions:

- How effective is the proposed framework, IAISLP, compared with other methods of link prediction?
- What are the effects of temporal information on the performance of link prediction?

### 4.1 Dataset

So far, microblogging has become one of important sources of information, as well as a main channel of information dissemination. Therefore, we leverage Sina microblog dataset [15] which contains time series of users' profile, topological structure and content information from September 28, 2012, to October 29, 2012 (one time stamp represents one day), to evaluate validity of the method we proposed. Statistics of the dataset are shown in Table 1.

**Table 1.** Statistics of the dataset.

# of nodes	# of edges	# of microblogs
1776950	308489739	300000

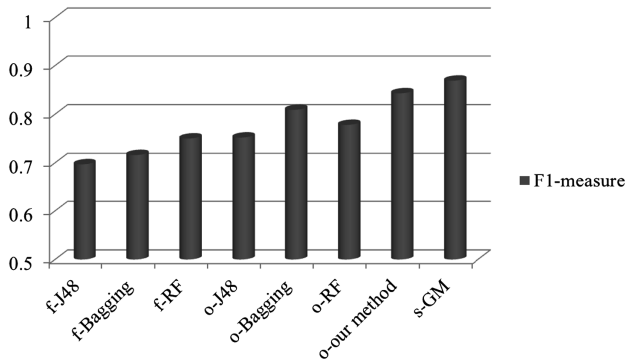
### 4.2 Performance Evaluation with Different Baseline Methods in Literature

To answer the first question, we compare IAISLP with following methods to verify the effectiveness of the proposed method.

- f-J48, f-Bagging and f-RF: Use J48 decision tree, Bagging and Random Forest to predict links based on the features defined in [11];
- o-J48, o-Bagging, o-RF: Use J48 decision tree, Bagging and Random Forest to predict links based on profile-based, relationship-based and emotion-based features proposed in this paper;
- o-IAISLP: Use IAISLP to predict links based on profile-based, relationship-based and emotion-based features proposed in this paper;
- s-GM: Since the features proposed in this paper don't apply to graph model, so in order to make a comparison with graph model, we use graph model to predict links based on the features defined in [12].

Figure 1 shows mean F1-measure for each method respectively.

From Fig. 1, the mean F1-measure of o-IAISLP can be increased by 9.6%, 3.4% and 6.5%, respectively, compared with o-J48, o-Bagging and o-RF: The mean F1-measure of o-J48 is the lowest on the features proposed in this paper since the information gain of J48 decision tree is biased toward those features with more numerical values. Bagging and Random Forest's mean F1-measures are not very different.



**Fig. 1.** Mean F1-measure of different link prediction methods.

Compared with other methods, IAISLP can achieve high mean F1-measure through deploying improved artificial immune system algorithm to predict links between users. In addition, the mean F1-measure of o-J48, o-Bagging and o-RF can be increased by 5.5%, 9.3% and 2.8%, respectively, compared with f-J48, f-Bagging and f-RF. It indicates that in addition to topological structure, the network is also flooded with massive user information which also affects users' behavior. Therefore, the features proposed in this paper can predict the existence and direction of links from a more comprehensive perspective. Finally, compared with s-GM, o-IAISLP shortens the execution time of link prediction ( $-33.6\%$ ) despite its mean F1-measure is slightly lower than s-GM ( $-2.6\%$ ). In summary, the proposed framework gains significant improvement over representative baseline methods, which answers the first question.

### 4.3 Analysis of Different Factors' Impacts in Link Prediction

To answer the second question, we also investigate how temporal information affects the performance of our method in terms of  $F1$ -measure by changing the time slice weight factor  $\alpha$ . In this paper,  $\alpha$  is varied as  $\{0.01, 0.2, 0.5, 0.7, 1\}$  and we carry on 10-fold cross-validations with 50%, 60%, 80%, and 100% of  $A$  for training so as to avoid bias brought by the sizes of the training data and the results are shown in Fig. 2, where "50%," "60%," "80%," and "100%" denote that we leverage 50%, 60%, 80%, and 100% of  $A$  for training.

It can be observed from Fig. 2: when setting  $\alpha$  as 1, namely, without considering temporal information, the  $F1$ -measure is much lower than the peak performance, and the  $F1$ -measure first increases greatly and then degrades rapidly after reaching a peak value with the increase  $\alpha$ .

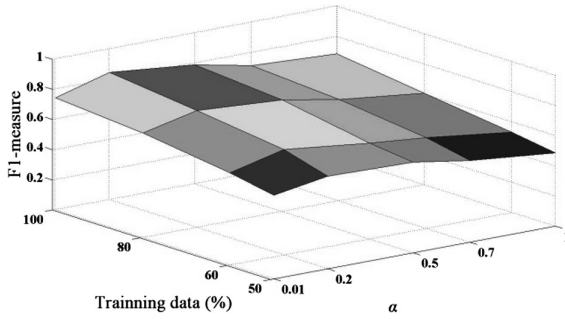


Fig. 2. The impact of temporal information in IAISLP.

## 5 Conclusion

Aiming at deficiencies of traditional link prediction algorithm, we put forward an improved artificial immune system model and deployed it to predict the existence and direction of link. Additionally, we ran a set of experiments to investigate the performance of our model, and reported system performances in terms of F1-measure which revealed that the proposed method can effectively improve performance of link prediction model.

## References

1. Barbieri, N., Bonchi, F., Manco, G.: Who to follow and why: link prediction with explanations. In: 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1266–1275. ACM, New York (2014)
2. Rozenshtein, P., Tatti, N., Gionis, A.: Inferring the strength of social ties: a community-driven approach. In: 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1017–1025. ACM, Halifax (2017)
3. Chen, Z., Chen, M., Weinberger, K.Q., Zhang, W.: Marginalized denoising for link prediction and multi-label learning. In: 29th AAAI Conference on Artificial Intelligence, pp. 1707–1713. AAAI Press, Austin (2015)
4. Wang, X., Hoi, S.C.H., Ester, M., Bu, J., Chen, C.: Learning personalized preference of strong and weak ties for social recommendation. In: 26th International Conference on World Wide Web, pp. 1601–1610. ACM, Perth (2017)
5. Kaewpitakkun, Y., Shirai, K.: Incorporating an implicit and explicit similarity network for user-level sentiment classification of microblogging. In: Booth, R., Zhang, M.-L. (eds.) PRICAI 2016. LNCS (LNAI), vol. 9810, pp. 180–192. Springer, Cham (2016). [https://doi.org/10.1007/978-3-319-42911-3\\_15](https://doi.org/10.1007/978-3-319-42911-3_15)
6. Zhang, Q., Liu, C., Zhong, S., Lei, K.: Spam comments detection with self-extensible dictionary and text-based features. In: 22nd IEEE Symposium on Computers and Communication, pp. 1225–1230. IEEE Computer Society, Heraklion (2017)
7. Lichtenwalter, R.N., Lussier, J.T., Chawla, N.V.: New perspectives and methods in link prediction. In: 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 243–252. ACM, Washington, DC (2010)

8. Wei, X., Xu, L., Cao, B., Yu, P.S.: Cross view link prediction by learning noise-resilient representation consensus. In: 26th International Conference on World Wide Web, pp. 1611–1619. ACM, Perth (2017)
9. Aggarwal, C., Subbian, K.: Evolutionary network analysis: a survey. *ACM Comput. Surv.* **47**(1), 1–36 (2014)
10. Zhang, J., Wang, C., Wang, J., Yu, P.S.: LaFT-tree: perceiving the expansion trace of one’s circle of friends in online social networks. In: 6th ACM International Conference on Web Search and Data Mining, pp. 597–606. ACM, Rome (2013)
11. Fire, M., Tenenboim-Chekina, L., Putis, R., Lesser, O., Rokach, L., Elovici, Y.: Computationally efficient link prediction in a variety of social networks. *ACM Trans. Intell. Syst. Technol.* **5**(1), 10 (2013)
12. Schall, D.: Link prediction in directed social networks. *Soc. Netw. Anal. Min.* **4**(1), 1–14 (2014)
13. Tang, M., Mao, X., Yang, S., Zhou, H.: A dynamic microblog network and information dissemination in “@” mode. *Math. Probl. Eng.* **15**, Article ID 492753 (2014)
14. Metcalf, K., Leake, D.B.: A computational method for extracting, representing, and predicting social closeness. In: 22nd European Conference on Artificial Intelligence, pp. 1176–1184. IOS Press, The Hague (2016)
15. Zhang, J., Liu, B., Tang, J., Chen, T., Li, J.: Social influence locality for modeling retweeting behaviors. In: 23rd International Joint Conference on Artificial Intelligence, IJCAI/AAAI, Beijing, China, pp. 2761–2767 (2013)



# Anchored Projection Based Capped $l_{2,1}$ -Norm Regression for Super-Resolution

Xiaotian Ma<sup>1</sup>, Mingbo Zhao<sup>1(✉)</sup>, Zhao Zhang<sup>2</sup>, Jicong Fan<sup>3</sup>,  
and Choujun Zhan<sup>4</sup>

<sup>1</sup> Donghua University, Shanghai, People's Republic of China  
mzhao4@dhu.edu.cn

<sup>2</sup> Soochow University, Suzhou, Jiansu, People's Republic of China

<sup>3</sup> City University of Hong Kong, Kowloon Tong, Hong Kong

<sup>4</sup> Nanfang College of Sun Yat-Sen University, Guangzhou, China

**Abstract.** Single image super resolution task is aimed to recover a high resolution image with pleasing visual quality from a single low resolution image. It is a highly under-constrained problem because of the ambiguous mapping between low/high resolution patch domain. In order to alleviate the ambiguity problem, we split input patches into numerous subclasses and collect exemplars according to the sparse dictionary atoms. However, we observe that there still exist some similar regressors do not share the same regression in the same subclass, which may increase the super-resolving error for training data in each cluster. In this paper, we propose a robust and effective method based capped  $l_{2,1}$ -norm regression to address this problem. The proposed method can automatically exclude outliers in each cluster during the training phase and give the potential to learn local prior information accurately. Numerous experimental results demonstrate that the proposed algorithm achieves better reconstruction performance against other state-of-the-art methods.

**Keywords:** Capped  $l_{2,1}$ -norm regression · Local linear regression  
Single image super resolution

## 1 Introduction

The goal of single image super resolution (SISR) technique is attempting to recover a high resolution (HR) image via a single low resolution (LR) image. The SISR task is a severely ill-posed problem, exploiting appropriate priors can guarantee the reconstruction stability. In recent years, learning-based methods have received a lot of attentions and many efficient methods are further developed. Assuming the missing high-frequency information lost in LR images can be learned from internal [1, 2] and external exemplars [3–12], these methods try to learn strong priors to ensure the uniqueness of reconstruction and improve SR performance.

Neighbor embedding methods are one of the representative learning-based methods for SISR reconstruction. Chang et al. [3] adopted the philosophy of

locally linear embedding (LLE) method and assumed an arbitrary HR patch can be linearly represented by its  $K$ -nearest neighbors based on the same weights which are computed in LR manifold. Since searching neighbors in the training pool of raw image patches are time-consuming and inefficient, Yang et al. [4] proposed to represent image patches by using sparse linear combination of atoms from an over-complete dictionary. In order to lower the computation, Zeyde et al. [5] reduced the dimension of image patches by adopting principle component analysis (PCA) approach. They then utilized K-SVD method to learn sparse dictionary pair effectively and adopted OMP method to accelerate sparse coding process. Bevilacqua et al. [6] further combined the theory of neighbor embedding and sparse coding, and proposed a non-negative neighbor embedding method, which selected  $K$  best neighbors from dictionary instead of raw patch database and proved the effectiveness when have the non-negative weights.

Recently, some efficient methods are to use local regression to learn mapping relation from LR input domain to HR domain. For example, Yang et al. [7] proposed to learn a collection of linear functions directly by separating LR\HR patches into numerous clusters. Timofte et al. [8] proposed an anchored neighbor regression (ANR) method, which can pre-calculate the corresponding embedding matrices offline to reduce execution time. Later, Jiang et al. [9] proposed an improved variant of ANR called LANR, which enforces the locality-constraint in the ridge regression in ANR. To further boost the reconstruction performance, Jiang et al. [10] exploited the non-local self-similarity based LANR method, and proposed LANR-NLM, which consists of a learning stage and a reconstruction stage. Moreover, Hu et al. [11] adopted a series of linear least squares functions named cascaded linear regression to compensate mapping residual progressively.

One challenge of regression-based SR methods is the inherently ambiguous problem between LR/HR domains. That is, many significant different HR patches may degrade to similar LR patches in the process of image down-sampling and the mapping between HR/LR data is many to one. Although dividing input patches into sufficient clusters can alleviate the ambiguity at certain extent, there still exist outliers which are in the same clusters do not share the same regression. In this paper, we propose an anchored projection based capped  $l_{2,1}$ -norm regression model to further alleviate the mapping ambiguity problem. That is to anchor the robust regression to the atoms of sparse dictionary and to precompute the corresponding projection matrices. Compared with ridge regression [7, 9, 10], our proposed  $l_{2,1}$ -norm regression is robust to handle outliers and make the learning process more stably. In addition, our proposed model is simple and effective, which has few parameters to control.

## 2 Related Work

### 2.1 Simple Functions for Super-Resolution

Simple functions [7] method is aimed to split LR/HR feature space into numerous subspaces and collect exemplars to learn priors for each subspace respectively. The authors adopt  $K$ -means method to cluster LR patches into a relatively

small number of subspaces, and attempt to utilize different functions for learning mapping: linear regression functions, and support vector regressors with a radial basis function kernel or a linear kernel. Due to the similar visual results, the authors proposed to utilize linear regressions as for its lower computation.

For each subspace, the problem can be addressed by the following:

$$V_i^*, b_i^* = \operatorname{argmin} \|V_i^T Y_i + b_i^T - X_i\|_2^2 \quad (1)$$

where  $V_i^*, b_i^*$  is the linear projection parameters,  $X_i$  and  $Y_i$  are the HR/LR training patches for cluster  $i$ ,  $X_i = \{x_1, x_2, \dots, x_{M_i}\}$ ,  $Y_i = \{y_1, y_2, \dots, y_{M_i}\}$ .

In online reconstruction phase, for each LR input  $y$  in cluster  $i$ , it can be super-resolved by applying the corresponding regression:

$$x = V_i^T y + b_i^T \quad (2)$$

## 2.2 Anchored Neighborhood Regression (ANR)

ANR approach [8] combines the sparse representation methods and the neighbor embedding methods. Starting from the same dictionary training strategy as Zeyde et al. [3]:

$$D^* = \operatorname{argmin} \|y - D\alpha\|_2^2, \quad \text{st. } \|\alpha\| \leq s, \quad (3)$$

where  $D$  is the LR dictionary,  $\alpha$  is the coefficients of sparse representation. ANR utilized the atoms of LR sparse dictionary as anchor points to associate the mapping function. By relaxing sparsity constraint from [4, 5], ANR reformulated Eq. (3) and adopted ridge regression to address this patch representation problem as follows:

$$\min_{\alpha} \|y - N_l \alpha\|_2^2 + \lambda \|\alpha\|_2^2 \quad (4)$$

where  $N_l$  is  $K$ -nearest neighbors from LR sparse dictionary. Then, assuming HR patches share the same coefficient on HR neighbors, it can be concluded that

$$x = N_h \alpha = N_h (N_l^T N_l + \lambda I)^{-1} N_l^T y = P_i y \quad (5)$$

where  $N_h$  is the HR neighbors corresponding to  $N_l$ ,  $P_i$  is the projection associated with anchor point  $d_i$ .

## 3 Proposed Method

### 3.1 Capped $l_{2,1}$ -Norm Regression

In order to alleviate ambiguous relation between HR/LR domain preliminarily, K-means method and dictionary method both are common ways for dividing the LR input patches. In our proposed method, we follow the works [8–10] to train a sparse LR dictionary and adopt the atoms  $d_i$  as the center to segment the input patches. Since the atoms of the sparse dictionary decomposed from a



high number of raw image patches, it can be efficiently representative for the whole dataset manifold [13]. Besides, instead of utilizing Euclidean distance to measure the similarity between patches, we consider to utilize the correlation metric. The segment scheme for each cluster  $d_i^*$  be formulated as

$$d_i^* = \operatorname{argmax} \operatorname{Corr}(y, d_i), \quad d_i \in D \quad (6)$$

For each cluster, we propose a robust regression method called capped  $l_{2,1}$ -norm regression. The formulation for each cluster can be expressed as:

$$P(V, b) = \min_{V_i, b_i} \sum_{j=1}^{M_i} \min(\|V_i^T y_j + b_i^T - X_j\|_2, \epsilon) \quad (7)$$

where  $V_i, b_i$  are the linear regression parameters,  $M_i$  is the total number of training samples in cluster  $i$  and  $\epsilon$  is the thresholding parameter to suppress the bias of outliers that are far away from the normal data distribution.

When we set a diagonal matrix  $W_i$  with the  $jj$ -th element as follows:

$$W_{i,jj} = \frac{1}{2} \|V_i^T y_j + b_i^T - x_j\|_2^{-1} \cdot \operatorname{Ind}(\|V_i^T y_j + b_i^T - x_j\|_2^2 \leq \epsilon) \quad (8)$$

where  $\operatorname{Ind}()$  is an indicative function, which is equal to 1 if  $\|V_i^T y_j + b_i^T - x_j\|_2 \leq \epsilon$  and 0 otherwise. Then we rewrite Eq. (7) as follows:

$$Q(V, b) = \min_{V, b} \sum_{j=1}^{M_i} W_{i,jj} \|V_i^T y_j + b_i^T - x_j\|_2^2 \quad (9)$$

The weight matrix  $W_i$  is used to exclude the outliers in each cluster  $i$  when we minimize the energy function. In other words, the outlier effects can be controlled by  $\epsilon$ . To solve this minimization of capped  $l_{2,1}$ -norm regression, we separate this objective function into two problems. First, we initialize the weight matrix  $W_i$  and set the derivatives of Eq. (9) w.r.t.  $V_i, b_i$  to zero. Here, we update  $V_i, b_i$  as:

$$\begin{aligned} V_i &= (Y L_c Y^T + \eta I)^{-1} Y L_c X^T \\ b_i &= (e W_i X^T - e W_i Y^T V) / e W_i e^T \end{aligned} \quad (10)$$

where  $e$  is a unit vector and  $L_c = W_i - W_i e^T e W_i / e W_i e^T$  is used for centering the samples by subtracting the mean of all samples. The parameter  $\eta$  is a small number tackles with the singular matrix. Then, with the parameters  $V_i, b_i$  fixed, the weight matrix  $W_i$  can be updated via Eq. (8). The iterative procedure will continue by iteratively update  $V_i, b_i$  and  $W_i$  until convergence.

### 3.2 Converge Analysis

In this subsection, we will prove the convergence of iterative process of regression parameters updating. Thus, our goals is to derive from  $Q(V_{(t+1)}, b_{(t+1)}) \leq Q(V_t, b_t)$  to prove the capped  $l_{2,1}$ -norm of objective function of Eq. (9) can

be monotonically decreased. Specifically, let  $K_t$  be the index of  $y_j$  that satisfies  $\|V_t^T y_j + b_t^T - x_j\|_2 \leq \epsilon$ , and  $|K_t|$  be the total number. Then, we rewrite  $Q(V_{(t+1)}, b_{(t+1)}) \leq Q(V_t, b_t)$  as follows:

$$\begin{aligned} \sum_{j \in K_t} W_{jj}^t \|V_{t+1}^T y_j + b_{t+1}^T - x_j\|_2^2 &\leq \sum_{j \in K_t} W_{jj}^t \|V_t^T y_j + b_t^T - x_j\|_2^2 \\ \sum_{j \in K_t} \frac{\|V_{t+1}^T y_j + b_{t+1}^T - x_j\|_2^2}{2\|V_t^T y_j + b_t^T - x_j\|_2} &\leq \sum_{j \in K_t} \frac{\|V_t^T y_j + b_t^T - x_j\|_2^2}{2\|V_t^T y_j + b_t^T - x_j\|_2} \end{aligned} \quad (11)$$

Following the property of inequality, i.e.  $\sqrt{a} - a/2\sqrt{b} \leq \sqrt{b} - b/2\sqrt{b}$  holds for any two positive real values, we have:

$$\begin{aligned} \sum_{j \in K_t} \left\{ \|V_{t+1}^T y_j + b_{t+1}^T - x_j\|_2 - \frac{\|V_{t+1}^T y_j + b_{t+1}^T - x_j\|_2^2}{2\|V_t^T y_j + b_t^T - x_j\|_2} \right\} \\ \leq \sum_{j \in K_t} \left\{ \|V_t^T y_j + b_t^T - x_j\|_2 - \frac{\|V_t^T y_j + b_t^T - x_j\|_2^2}{2\|V_t^T y_j + b_t^T - x_j\|_2} \right\} \end{aligned} \quad (12)$$

Combining Eqs. (11) and (12) and adding  $(M_i - |K_t|)\epsilon$  to both sides of inequality:

$$\sum_{j \in K_t} \|V_{t+1}^T y_j + b_{t+1}^T - x_j\|_2 + (M_i - |K_t|)\epsilon \leq \sum_{j \in K_t} \|V_t^T y_j + b_t^T - x_j\|_2 + (M_i - |K_t|)\epsilon \quad (13)$$

Therefore, if we let  $K_{(t+1)}$  be the index of  $y_j$  that satisfies  $\|V_{(t+1)}^T y_j + b_{(t+1)}^T - x_j\|_2 \leq \epsilon$ , and  $|K_{(t+1)}|$  be the total number, then no matter whether  $|K_{(t+1)}| \leq |K_t|$  or  $|K_{(t+1)}| > |K_t|$ , it follows

$$\begin{aligned} \sum_{j \in K_{t+1}} \|V_{t+1}^T y_j + b_{t+1}^T - x_j\|_2 + (M_i - |K_{t+1}|)\epsilon \\ \leq \sum_{j \in K_t} \|V_{t+1}^T y_j + b_{t+1}^T - x_j\|_2 + (M_i - |K_t|)\epsilon \end{aligned} \quad (14)$$

Combined Eqs. (13) and (14), we have

$$\sum_{j=1}^{M_i} \min(\|V_{t+1}^T y_i + b_{t+1}^T - x_j\|_2, \epsilon) \leq \sum_{j=1}^{M_i} \min(\|V_t^T y_j + b_t^T - x_j\|_2, \epsilon) \quad (15)$$

which prove  $P(V_{(t+1)}, b_{(t+1)}) \leq P(V_t, b_t)$ .

## 4 Experimental Results

### 4.1 Experimental Settings

In our experiments, we only perform SR on the illuminance (Y) channel, since human vision is more sensitive to intensity changes than color variations. In order

**Table 1.** Average evaluation on Set5, Set14, BSD100 with scale factor x3, x4

Methods	Dataset					
	Set5		Set14		BSD100	
	x3	x4	x3	x4	x3	x4
Bicubic	26.48/0.788	24.75/0.732	24.72/0.679	24.72/0.679	25.13/0.648	24.03/0.596
Zeyde [5]	31.20/0.882	29.30/0.837	28.11/0.784	26.67/0.725	27.52/0.745	26.36/0.686
SF [7]	32.11/0.899	29.89/0.852	28.74/0.804	27.10/0.740	27.99/0.766	26.62/0.700
ANR [8]	31.45/0.886	29.48/0.842	28.27/0.792	26.77/0.732	27.66/0.755	26.44/0.694
LANR [9]	31.85/0.894	29.63/0.844	28.61/0.804	26.87/0.735	27.92/0.769	26.52/0.698
LANR_NLM [10]	31.93/0.896	29.64/0.845	28.71/0.804	26.87/0.737	27.99/0.775	26.53/0.700
Ours	32.32/0.904	29.93/0.856	28.90/0.810	27.16/0.743	28.11/0.773	26.65/0.702

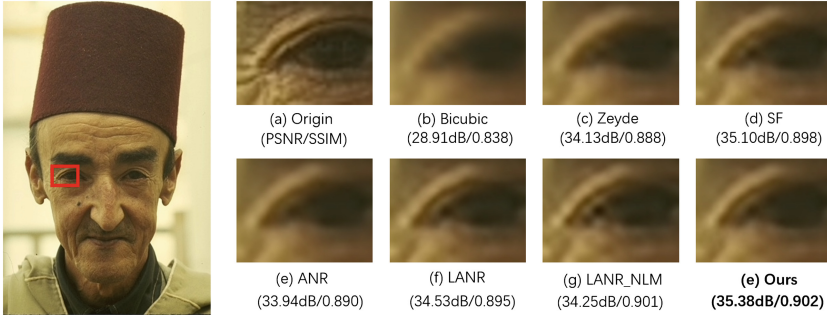
to get an intuitive representation from LR input image, we adopt first-order and second-order gradients on horizontal and vertical directions to extract the high-frequency for LR features. The image patch size is set to  $9 \times 9$ ,  $12 \times 12$  for different upscaling factors x3, x4, respectively. And the overlap size between neighbor patches is related with the patch size and the magnification factor, which is set to 6, 8 pixels respectively. The LR input images used in our experiments are followed by [8], which are downsampled from original HR images and blurred by a  $7 \times 7$  Gaussian blurring operator with a standard deviation of 1.6.

## 4.2 Parameters Analysis

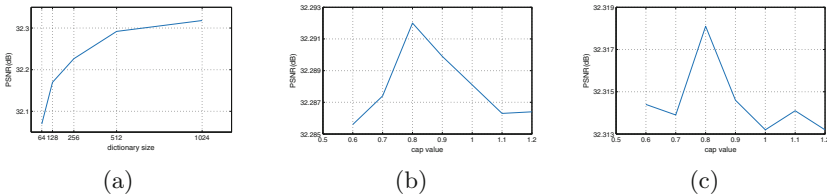
In this subsection, we mainly analyze the influence of the number of clusters  $T$  and the value of the cap  $\epsilon$ , which play an important role in our model.

*Influence of the Cluster Number  $T$ :* In Fig. 2(a), we have shown the average PSNR performance on Set5 dataset according to various cluster number. We can observe that the reconstruction performance improves rapidly with the cluster number  $T$  increasing, since dividing LR input patches into sufficient can alleviate the ambiguity problem to some extent. However, a large number of clusters can slow down the reconstruction speed since it needs more time to traverse all the cluster center to apply the corresponding regression. In this paper, we set 1024 clusters in our experiments.

*Influence of the Cap Value  $\epsilon$ :* The cap  $\epsilon$  is used for identifying the outliers and further alleviate the ambiguity problem in training process. In Fig. 2(b),(c), we present the average PSNR of Set5 test dataset according to various cap value, with different clusters. It is notable that the litter  $\epsilon$  is, the stronger constraint imposed for outliers. However, when  $\epsilon$  is too small, it will filtrate amounts of normal data and some clusters may not get enough training samples to model the local regression, which tends to cause over-fitting. On the other hand, when  $\epsilon$  is set to a large number, our proposed capped  $l_{2,1}$ -norm regression will degrade to  $l_{2,1}$ -norm regression. Therefore, in order to learn accurate mapping function at the greatest extent, we set the cap value as 0.8 in our experiments.



**Fig. 1.** Visual quality comparisons with different SR methods on “189080” from BSD100 dataset (upscaling factor = x3)



**Fig. 2.** Parameter analysis of our proposed method. (a) Influence of dictionary size. (b) Influence of cap value with 512 dictionary size. (c) Influence of cap value with 1024 dictionary size.

### 4.3 Compared Methods

In this subsection, we adopt peak signal to noise ratio (PSNR) and structure similarity (SSIM) to evaluate the objective quality of our model and other state-of-the-art methods. Taking bicubic method as a baseline, we compare our proposed method with Zeyde [5], SF [7], ANR [8], LANR [9], and LANR\_NLM [10] models on objective metrics in Table 1. For magnification factor x3 or x4, we both find that our proposed method achieves an obvious higher average PSNR and SSIM than other methods. Compared with several local regression methods, such as SF [7], ANR [8], LANR [9], our proposed method also shows competitive performance without sacrificing reconstruction speed. We attribute these better results to robust local regression learning.

The visual quality with different methods are shown in Fig. 1. As we can see, our proposed method shows the cleaner and shaper edges and generate more competitive quality. Although LANR, LANR\_NLM methods show better performance in recovering high-frequency information, but they also introduce the artifacts around edges. The experiment results show that our proposed method achieves pleasing results with fewer artifacts.

## 5 Conclusion

In this paper, we proposed a novel local linear regression for SISR based capped  $l_{2,1}$ -norm function. By splitting LR input patches into numerous subclasses and learning a series of robust linear regression, the proposed method shows pleasing visual quality with sharp edges. In addition, the projection matrices used during target reconstruction all can be learned offline, which can reduce huge execution time. Results on benchmark datasets demonstrate the effectiveness in term of qualitative and quantitative performance of our proposed method.

**Acknowledgment.** This work is supported by the National Science Foundation of China under Grant no. 61601112, the Fundamental Research Funds for the Central Universities and DHU Distinguished Young Professor Program.

## References

1. Bevilacqua, M., Roumy, A., Guillemot, C., Alberi-Morel, M.L.: Low-complexity single-image super-resolution based on nonnegative neighbor embedding (2012)
2. Chang, H., Yeung, D.Y., Xiong, Y.: Super-resolution through neighbor embedding. In: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2004, vol. 1, p. I. IEEE (2004)
3. Glasner, D., Bagon, S., Irani, M.: Super-resolution from a single image. In: 2009 IEEE 12th International Conference on Computer Vision, pp. 349–356. IEEE (2009)
4. Gu, S., Sang, N., Ma, F.: Fast image super resolution via local regression. In: 2012 21st International Conference on Pattern Recognition (ICPR), pp. 3128–3131. IEEE (2012)
5. Hu, Y., Wang, N., Tao, D., Gao, X., Li, X.: SERF: a simple, effective, robust, and fast image super-resolver from cascaded linear regression. *IEEE Trans. Image Process.* **25**(9), 4091–4102 (2016)
6. Huang, J.B., Singh, A., Ahuja, N.: Single image super-resolution from transformed self-exemplars. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5197–5206 (2015)
7. Jiang, J., Fu, J., Lu, T., Hu, R., Wang, Z.: Locally regularized anchored neighborhood regression for fast super-resolution. In: 2015 IEEE International Conference on Multimedia and Expo (ICME), pp. 1–6. IEEE (2015)
8. Jiang, J., Ma, X., Chen, C., Lu, T., Wang, Z., Ma, J.: Single image super-resolution via locally regularized anchored neighborhood regression and nonlocal means. *IEEE Trans. Multimed.* **19**(1), 15–26 (2017)
9. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: 2001 Proceedings of the Eighth IEEE International Conference on Computer Vision, ICCV 2001, vol. 2, pp. 416–423. IEEE (2001)
10. Timofte, R., De Smet, V., Van Gool, L.: Anchored neighborhood regression for fast example-based super-resolution. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1920–1927 (2013)
11. Yang, C.Y., Yang, M.H.: Fast direct super-resolution by simple functions. In: 2013 IEEE International Conference on Computer Vision (ICCV), pp. 561–568. IEEE (2013)

12. Yang, J., Wright, J., Huang, T.S., Ma, Y.: Image super-resolution via sparse representation. *IEEE Trans. Image Process.* **19**(11), 2861–2873 (2010)
13. Zeyde, R., Elad, M., Protter, M.: On single image scale-up using sparse-representations. In: Boissonnat, J.-D., et al. (eds.) *Curves and Surfaces 2010*. LNCS, vol. 6920, pp. 711–730. Springer, Heidelberg (2012). [https://doi.org/10.1007/978-3-642-27413-8\\_47](https://doi.org/10.1007/978-3-642-27413-8_47)



# Query Expansion Based on Semantic Related Network

Limin Guo<sup>1</sup>, Xing Su<sup>1</sup>, Ling Zhang<sup>2(✉)</sup>, Guangyan Huang<sup>3</sup>, Xu Gao<sup>4</sup>,  
and Zhiming Ding<sup>1</sup>

<sup>1</sup> Faculty of Information, Beijing University of Technology, Beijing 100124, China  
guolimin@bjut.edu.cn

<sup>2</sup> National Earthquake Response Support Service, Beijing 100049, China  
zhangling903@163.com

<sup>3</sup> School of Information Technology, Deakin University, Melbourne, Australia

<sup>4</sup> Smart City Institute, Zhengzhou University, Zhengzhou 450001, Henan, China

**Abstract.** With the development of big data, the heuristic query based on the semantic relationship network has become a hot topic, which attracts much attention. Due to the complex relationship between data records, the traditional query technologies cannot satisfy the requirements of users. To this end, this paper proposes a heuristic query method based on the semantic relationship network, which first constructs the semantic relationship model, and then expands the query based on the constructed semantic relationship network. The experiments demonstrate the reasonableness, high precision of our method.

**Keywords:** Semantic relation · Domain ontology  
Semantic relationship network · Query expansion · Heuristic query

## 1 Introduction

With the rapid development of information communication technologies and related applications, various sources of data expand rapidly, such as web data, social network data, traffic data, IoT data, etc., which have greatly changed people's life style. The traditional query technology is hard to satisfy the user's information retrieval requirements. Therefore, how to effectively use these data for query service is one of the major challenges faced by researchers. In this paper, we study the heuristic query expansion method, which extends the query words semantics and searches the related data objects.

Heuristic query expansion has great research values and wide applications. For example, in the field of emergency rescue, the user can derive similar disasters, rescue plans, distribution of medical, rescue force near the disaster area from querying a given disaster, which can play a positive guiding role in the rescue; in the field of e-commerce, the relationship of user shopping behaviors can be analyzed from user's purchase data, which can be used to recommend items of interest to them. So, more and more attention has been paid to heuristic query expansion on the semantic relationship.

In this paper, we propose a heuristic Query Expansion based on the Semantic Related network (short for QEO SR). QEO SR has the advantages that (1) it builds a semantic

related network with related weights, and adjusts the related network dynamically, which can provide more accurate and effective search structure, and improve the retrieval quality effectively; (2) it gets heuristic query expansion through the semantic related network, and improves the accuracy effectively. The main contributions of this paper are summarized as: (1) We define a semantic related network model with association weights; (2) We construct the semantic related network based on temporal weights; (3) We extend the query based on the semantic related network; (4) We demonstrate the reasonableness and high precision of QEOsR.

## 2 Related Work

Generally speaking, query expansion methods can be classified into three categories: query expansion based on global and local analysis, query expansion based on user query log and query expansion based on the semantic network.

Query expansion based on global and local analysis finds the similarity between words according to the co-occurrence information of the document to realize the query extension [1, 2]. Query expansion based on query log builds the relationship between the query space and document space by analysing user’s query log to extend query [3, 4]. These methods mainly focus on the synonym extension that keep the synonymity of expansion words and query words, and the synonymity of extended query and original query, while ignoring the semantic relationship between concepts. Therefore, the user’s real intension cannot be expressed fundamentally.

Query expansion based on the semantic network [5–8] extends query from the semantic concept level, which mainly uses the semantic network to compute similarity and realize query expansion [9–11]. [12] is the most related work to this paper, which proposes a two-stage query expansion method based on the semantic network. Specifically, a semantic network, e.g. ConceptNet, is used to select the concepts from each layer in a one-by-one manner while maintaining the desired level of precision and minimizing the number of concepts that need to be examined. However, the above method has following limitations: (1) ConceptNet does not consider the temporal factors which cannot dynamically adjust the concept graph; (2) The extended query may break the relationship between layers and lose the related paths between expansion words.

The above methods are studied from statistical or semantic perspectives. However, the temporal factor is not taken into account, and the complete related path between expansion words is neglected. In order to solve the above problems, this paper proposes a method of heuristic query expansion based on the semantic related network.

## 3 Problem Description and Definitions

**Definition 1 (Semantic Related Graph):** A semantic related graph  $RG$  is defined as:

$$RG = (V, E),$$



where  $V$  is a set of nodes,  $\forall v \in V$ ,  $v$  represents a concept, which is denoted by annotation  $A$ ;  $E$  is a set of edges,  $\forall e \in E$ ,  $e = (v_i, v_j, rw)$  represents a relationship between  $v_i$  and  $v_j$ , and the related weight is  $rw$ . For convenience, the semantic related graph is called related graph for short.

**Definition 2 (Query):** A query  $Q$  consists of a set of annotations, which is defined as:

$$Q = (A_1, A_2, \dots, A_m),$$

where  $A_i$  represents the concept corresponding to the  $i^{\text{th}}$  query condition.

**Problem Statement:** Given a query  $Q$ , the problem is to implement the heuristic query expansion, named  $Expansion(Q)$ . The extension steps are described as follows: (1) On the basis of corpus text set, we construct a semantic related graph  $RG$  based on the concept set of domain ontology; (2) On the basis of  $Q$  and  $RG$ , we extend the query through heuristic related search to get extended query  $Q'$ .

Therefore, we propose a query expansion method, QEOsR, based on the semantic related network. The general framework of QEOsR can be divided into two parts: (1) the semantic related network construction and (2) the heuristic query expansion, which is shown in Fig. 1. We first propose a method to construct the semantic related network based on temporal weights. Specifically, on the basis of corpus text set, we construct a semantic related network based on the concept set of domain ontology; Then, we analyze the query keywords and extend the query based on the semantic related network.

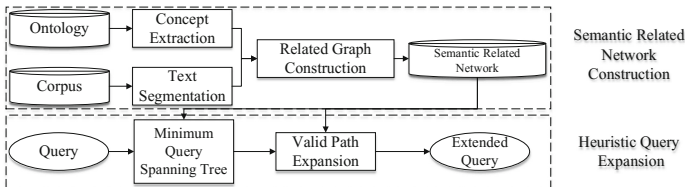


Fig. 1. System framework of QEOsR

## 4 Semantic Related Network Construction

Before the construction of the related graph, we should do word segmentation, syntax analysis and semantic analysis of the text in the corpus to generate one or more semantic tags firstly, which have been studied in many literatures [13–15]. The details of this phase are omitted, due to space limitation. Then, we construct the related graph based on the obtained tags and the concepts in the domain ontology.

Generally speaking, if two concepts co-occur frequently in the text, we can assume that there is a semantic connection between the two concepts, and the higher the frequency of co-occurrence, the stronger the relevance between them.

**Definition 3 (Co-occurrence Rate):** Given a text set  $ST = (st_1, st_2, \dots, st_n)$ , and two annotations  $A_1$  and  $A_2$ , the co-occurrence rate of  $A_1$  and  $A_2$  is defined as:

$$CP(A_1, A_2) = \frac{\sum_{i=1}^n f_i(A_1, A_2)}{n},$$

where  $f_i(A_1, A_2) = \begin{cases} 1 & A_1, A_2 \text{ co-occur in } st_i \\ 0 & \text{otherwise} \end{cases}$ , which represents the co-occurrence of  $A_1$  and  $A_2$  in  $st_i$ .

The co-occurrence rate reveals the probability of simultaneous occurrence of two annotations. However, it cannot accurately reflect the relevance between the two annotations, because the relevance is also affected by the respective occurrence probabilities of the two annotations. Therefore, we give the definition of the related degree.

**Definition 4 (Related Degree):** Given a text set  $ST = (st_1, st_2, \dots, st_n)$ , and two annotations  $A_1$  and  $A_2$ , the related degree of  $A_1$  and  $A_2$  is defined as follows:

$$RP(A_1, A_2) = \frac{CP(A_1, A_2)}{\sqrt{OP(A_1) \cdot OP(A_2)}},$$

where  $CP(A_1, A_2)$  is the co-occurrence rate of  $A_1$  and  $A_2$  in  $ST$ , and  $OP(A_i)$  is the occurrence probability of  $A_i$  in  $ST$ .

The related degree adopts the idea of Salton [16], which reveals the probability that two annotations appear simultaneously, when they appear respectively.

After the related graph is constructed, it will change because the relationship between concepts varies over time. Suppose that time is divided into a series of time slots  $T = \{t_1, t_2, \dots, t_n\}$ , and the incremental texts will be collected and put into corpus periodically at each time slot. We first compute the co-occurrence rates and the related degrees between concepts in the current time slot  $t_i$ , and save them in the related matrix  $RM_{it}$ . After that, we combine the historically related matrix to dynamically adjust the related graph.

Algorithm 1 presents the pseudo code of the related matrix computing. We first extract the conceptual set  $AS$  of the specific domain from ontology (line 1). Then, we scan the text set to see if each concept appears and initialize matrix  $OM$ , where each line represents a text, each column represents a concept, and  $OM[i, j]$  indicates whether the  $j^{\text{th}}$  concept appears in the  $i^{\text{th}}$  text (lines 2–7). After that, we calculate the co-occurrence rates and the related degrees of each concept pair in the text set, select the strong relevant concept pairs, and store them in the related matrix  $RM$  (line 8–17).

**Algorithm 1: Related Matrix Computing**

**Input:**  $ST$ : a text set,  $Ontology$ : an ontology,  $min\_cp$ : minimum co-occurrence rate,  $min\_rp$ : minimum related degree

**Output:**  $RM$ : a related matrix

```

1.  $AS \leftarrow \text{ExtractConcept}(Ontology)$ ; //Extract the conceptual set  $AS$ 
2. foreach  $st_i \in ST$  do //Initialize maxtrix  $OM$ 
3.   foreach  $A_j \in AS$  do
4.     if  $A_j$  is in  $st_i$  then
5.        $OM[i, j] \leftarrow 1$ ;
6.     else
7.        $OM[i, j] \leftarrow 0$ ;
8.   foreach  $A_j \in AS$  do //Compute the related matrix  $RM$ 
9.     foreach  $A_j \in AS$  do
10.       $CP(A_i, A_j) \leftarrow \sum_{k=1}^{|ST|} OM[k, i] \times OM[k, j] / |ST|$ ;
11.       $OP(A_i) \leftarrow \sum_{k=1}^{|ST|} OM[k, i] / |ST|$ ,  $OP(A_j) \leftarrow \sum_{k=1}^{|ST|} OM[k, j] / |ST|$ ;
12.       $RP(A_i, A_j) \leftarrow CP(A_i, A_j) / \sqrt{OP(A_i) \times OP(A_j)}$ ;
13.      if  $CP(A_i, A_j) \geq min\_cp$  &&  $RP(A_i, A_j) \geq min\_rp$  then
14.         $RM[i, j].cp \leftarrow CP(A_i, A_j)$ ,  $RM[i, j].rp \leftarrow RP(A_i, A_j)$ ;
15.      else
16.         $RM[i, j].cp \leftarrow 0$ ,  $RM[i, j].rp \leftarrow 0$ ;
17. return  $RM$ ;

```

Suppose the current time slot is  $t_k$ , for any time slot  $t_l$  ( $l \leq k$ ), the time weight of related degree between concept pair is  $W_T(l, k) = 1/2^{(k-l)}$ . Given a window  $Wd = \{t_{k-n}, t_{k-n+1}, \dots, t_k\}$ , the formula of the related weight of any concept pair  $(A_i, A_j)$  is described as follows.

$$rw(A_i, A_j) = 1 - \frac{1}{|Wd|} \sum_{t_l \in Wd} W_T(l, k) RP_{t_l}(A_i, A_j), \quad (3)$$

where  $RP_{t_l}(A_i, A_j)$  is the related degree of  $(A_i, A_j)$  at the time slot  $t_l$ , the related weight of  $(A_i, A_j)$  is the averagely weighted sum of the time weights of the related degrees within the window  $Wd$ , and the closer to the current time, the greater the time weight.

It can be seen from Formula (3), that the smaller the related weight between concepts, the more relevant relationship between concepts.

Algorithm 2 shows the related graph construction algorithm. We first take the concept in the conceptual set as the node of the related graph, and initialize the related weight of each concept pair (lines 1–3). Then, we extract the related matrix set  $RMS'$  in the window  $Wd$  (line 4). Next, the time weights of the related degree between nodes are calculated (lines 5–8). After that, the related weights between nodes are calculated according to Formula (3), if there exists a strong relationship, then add an edge (lines 9–12).

**Algorithm 2: Related Graph Construction**

**Input:**  $ST$ : a text set,  $AS$ : a conceptual set,  $RMS$ : a related matrix set,  $Wd$ : a window,  $t_k$ : the current time slot

**Output:**  $RG$ : a related graph

```

1.  $V \leftarrow AS, E \leftarrow \emptyset$ ;
2. foreach pair  $(A_i, A_j)$  in  $AS$  do
3.    $rw(A_i, A_j) \leftarrow 0$ ;
4.  $RMS' \leftarrow \text{ExtractRM}(RMS, Wd)$ ;
5. foreach  $RM_{ll} \in RMS'$  do
6.   foreach  $i: 1$  to  $|AS|-1$  do
7.     foreach  $j: i+1$  to  $|AS|$  do
8.        $rw(A_i, A_j) \leftarrow rw(A_i, A_j) + W_T(l, k)RM_{ll}[i, j].cp$ ;
9.   foreach pair  $(A_i, A_j)$  in  $AS$  do
10.     $rw(A_i, A_j) \leftarrow 1 - \frac{1}{Wd} rw(A_i, A_j)$ ;
11.    if  $rw(A_i, A_j) < 1$  then //There is a relationship between  $A_i$  and  $A_j$ 
12.       $E \leftarrow E \cup (A_i, A_j, rw(A_i, A_j))$ ;
13. return  $RG(V, E)$ ;
```

## 5 Heuristic Query Expansion

Heuristic query is an extended query on relevant semantics, which can be achieved by the semantic related graph. Given a query  $Q = (A_1, A_2, \dots, A_m)$ , we should find out all strong relevant semantic expansion to provide more support for the query.

Intuitively, the stronger relevant to the query words are, the closer relationship between them is. That is to say, the concepts that have less related weights with query words would have a closer relationship with them. Thus, the problem can be transformed to find a subgraph in  $RG$  so that all the query words in  $Q$  are covered, and not only all query words are reachable, but also the total related values of the path is minimal, then it can be considered that the subgraph is the most relevant extended subgraph with  $Q$ .

**Definition 5 (Query Spanning Tree):** Given a related graph  $RG$ , and a query  $Q$ , a query spanning tree  $T$  satisfied the following conditions:

- (1)  $T$  contains each node in the  $Q$ ;
- (2)  $T$  does not contain rings;
- (3)  $T$  is a connected subgraph of  $RG$ .

The related value of the query spanning tree  $T = (V', E')$  is represented as  $W(T) = \sum_{e_i \in E'} rw(e_i)$ , where  $rw(e_i)$  is the related weight of edge  $e_i$ .

**Definition 6 (Minimum Query Spanning Tree):** Given a related graph  $RG$ , and a query  $Q$ , the minimum query spanning tree  $MQST$  is a query spanning tree, and it satisfies the following condition:

$$W(MQST) = \min \{W(T) | T \in TS\},$$

where  $TS = \{T_1, T_2, \dots, T_n\}$  is a set of all query spanning trees that satisfy  $RG$  and  $Q$ .

In fact, the minimum query spanning tree is the query spanning tree with minimum related value.

Therefore, in the process of query expansion, we first construct a minimum query spanning tree based on the query  $Q$  and the related graph  $RG$ , and then further expand  $Q$ .

The Prim algorithm [17] is an effective method to build a minimum spanning tree. However, the minimum spanning tree is different from the minimum query spanning tree. First, the nodes of the former are all nodes in the graph, while the nodes of the latter are the all nodes in the query; Second, the former has  $n-1$  edges ( $n$  is the number of nodes), while the number of edges is uncertain in the latter. We modify the Prim algorithm in the literature [17], which uses matrix  $M$  to store the shortest path and related value between each pair of nodes in the query, and uses adjacency list  $adj$  to save the connectivity of each node in the query.

---

**Algorithm 3: Minimum query spanning tree**


---

**Input:**  $Q$ : a query,  $RG$ : a related graph

**Output:**  $MQST$ : a minimum query spanning tree

---

```

1.  $v_0 \leftarrow \text{Random}(Q)$ ,  $V' \leftarrow v_0$ ,  $E' \leftarrow \emptyset$ ,  $adj[\ ] \leftarrow 0$ ;
2. foreach  $u \in Q$  do
3.   foreach  $v \in Q$  do
4.     if  $(u, v)$  is connected then //Path reachable between  $u$  and  $v$ 
5.        $adj[u] \leftarrow adj[u] \cup v$ ;
6.        $M[u, v].sp \leftarrow \text{ShortPath}(RG, u, v)$ ,  $M[u, v].w \leftarrow W(M[u, v].sp)$ ;
7.     else
8.        $M[u, v].sp \leftarrow \text{Null}$ ,  $M[u, v].w \leftarrow \infty$ ;
9.   foreach  $u \in Q$  do
10.     $weight[u] \leftarrow \infty$ ;
11.  $weight[v_0] \leftarrow 0$ ,  $Heap \leftarrow Q$ ;
12. while  $Heap \neq \emptyset$  do
13.    $u \leftarrow \text{PopMin}(Heap)$ ;
14.   foreach  $v \in adj[u]$  do
15.     if  $M[u, v].w < weight[v]$  then
16.        $weight[v] \leftarrow M[u, v].w$ ;
17.        $adjust(Heap)$ ;
18.        $path(v_1, v_2, \dots, v_m) \leftarrow M[u, v].sp$ ;
19.       foreach  $v_i \in path$  do
20.          $V' \leftarrow V' \cup v_i$ ;
21.       foreach  $(v_i, v_j) \in path$  do
22.          $E' \leftarrow E' \cup (v_i, v_j, rw(v_i, v_j))$ ;
23. return  $MQST(V', E')$ ;

```

---

In Algorithm 3, we constantly choose a new path from  $RG$  and add it to the  $MQST$ , which makes the related value of  $MQST$  minimal, that is, the most relevant with  $Q$ . We first randomly choose a node as the start node from  $Q$ , and initializes the node set  $V'$  and the edge set  $E'$  in  $MQST$  (lines 1). Then, for each pair of concept  $(u, v)$  in  $Q$ , we decide whether its connectivity, if so, the connected nodes would be stored in  $adj$ , the shortest path between  $(u, v)$  and their related value would be saved in  $M[u, v].sp$  and  $M[u, v].w$  respectively; Otherwise, they would be set to empty and infinity respectively (lines 2–8). After that, we initialize the weight array  $weight$  and the min heap  $Heap$ , where  $weight[v]$  represents the minimum related value in the related value of all paths from  $v$  to  $MQST$  (lines 9–11). Next, we start to iterate, pop the minimum node  $u$  in  $weight$  from  $Heap$ , and the value of  $u$ 's each connectivity node  $v$  of  $Q$  is updated in  $weight$ , which ensures it is the minimum related value from  $v$  to  $MQST$ ; Then, we adjust the

heap and add the nodes and edges in the path to set  $V'$  and  $E'$  until the heap is empty, that is, the nodes in  $Q$  are all added to  $MQST$ , the iteration is finished (lines 12–22).

**Definition 7 (Valid Path):** Given a node  $v$ , a related value threshold  $\delta$ , and a layer threshold  $\tau$ ,  $VP = (v'_1, v'_2, \dots, v'_k)$  is the valid path of  $v$  if and only if the followings are satisfied:

$$(1) v = v'_1; \quad (2) W(VP) \leq \delta; \quad (3) k \leq \tau.$$

The nodes in the query  $Q$  should be further extended, the **heuristic query expansion** of  $Q = (v_1, v_2, \dots, v_m)$  is  $Q' = MQST \cup VPS_{v_1} \cup VPS_{v_2} \cup \dots \cup VPS_{v_m}$ , where  $VPS_{v_i}$  represents the valid path set of  $v_i$ .

## 6 Experimental Results

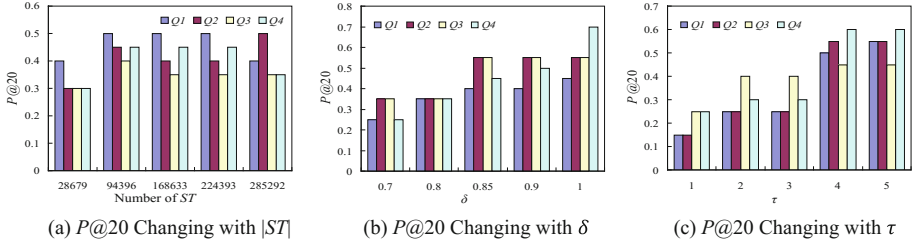
We choose 500 web pages on the Internet as experimental data, which are retrieved the news of “emergency rescue” from Baidu. By dividing these pages according to the window, 285292 segments can be extracted and put into the corpus. First, we pre-process the texts in the corpus, including Chinese word segmentation, semantic tag tagging and dividing the time slot by day. Then, we extract the conceptual set based on HowNet ontology dictionary, and dynamically construct the semantic related network. On this basis, we design some queries and extend them by QEO SR. Table 1 gives the query list.

**Table 1.** Query list

Query Id	Query words
$Q_1$	(Accident, Emergency, Accident Type, Emergency Evacuation)
$Q_2$	(Accident, Fire Station, Rescue Channel, Medical Treatment)
$Q_3$	(Accident, Fire Station, Rescue Channel, Hospital)
$Q_4$	(Accident, Command, Person, Coordinate)

### 6.1 Accuracy

In order to better evaluate the accuracy of QEO SR query expansion method, we choose the most commonly used evaluation indicator  $P@20$  in the information retrieval field to measure the accuracy. Figure 2 shows  $P@20$  with different parameters.

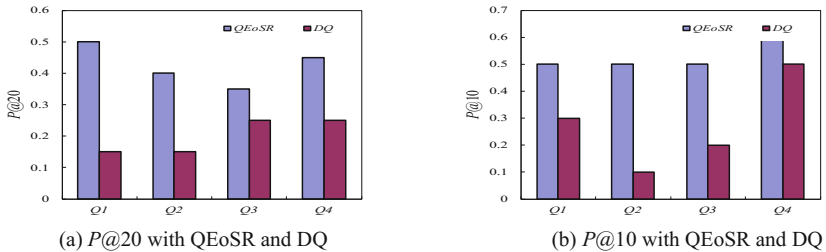


**Fig. 2.** Accuracy evaluation

From Fig. 2(a) we can see that there is no obvious rule between accuracy and the number of text set size. Because the experimental data is the relevant texts in the field of emergency rescue, where the relationship between the conceptual words is relatively close. So the connection between the concepts is not directly related to the number of text set. From Fig. 2(b) we can see that with the increase of the related value threshold, the accuracy of extended query increases, since the related value threshold increases, the query extension words also increase. Correspondingly, the accuracy increases. Similarly with the related threshold, the accuracy of the extended query increases with the increase of the layer threshold, which is shown in Fig. 2(c). Because the query expansion words increases as the layer threshold increases, so does the accuracy.

## 6.2 Effectiveness

To illustrate the effectiveness of the heuristic query expansion method, we compare the results of query expansion (QEoSR) and the results of the direct query (DQ). Figure 3 shows  $P@20$  and  $P@10$  of different queries with QEoSR and DQ respectively.



**Fig. 3.** Effectiveness evaluation

From Fig. 3 we can see that with the increase of the number of query expansion words, QEoSR has obvious advantages in accuracy, which is much better than DQ.

## 7 Conclusion

In this paper, we introduce the concept of semantic related graph, and propose a heuristic query expansion method, called QEoS<sub>R</sub>, based on the related graph. The general framework of QEoS<sub>R</sub> is divided into two parts: semantic related network construction and heuristic query expansion. Experimental results demonstrate the reasonableness and high accuracy of QEoS<sub>R</sub>.

**Acknowledgement.** This work is supported by National Key R&D Program of China (No. 2017YFC0803300), the National Natural Science of Foundation of China (No. 61703013, 91546111, 91646201) and the Key Project of Beijing Municipal Education Commission (No. KM201810005023, KM201810005024, KZ201610005009).

## References

1. Limsopatham, N., Macdonald, C., Ounis, I.: Modelling the usefulness of document collections for query expansion in patient search. In: *CIKM 2015*, pp. 1739–1742 (2015)
2. Xu, Y., Jones, G.J.F., Wang, B.: Query dependent pseudo-relevance feedback based on Wikipedia. In: *SIGIR 2009*, pp. 59–66 (2009)
3. Oliverira, V., Gomes, G., Blem, F., et al.: Automatic query expansion based on tag recommendation. In: *CIKM 2012*, pp. 1985–1989 (2012)
4. Cui, H., Wen, J., Li, M.: A statistical query expansion model based on query logs. *J. Softw.* **14**(9), 1593–1599 (2003). (in Chinese)
5. Bakhtin, A., Ustinovskiy, Y., Serdyukov, P.: Predicting the impact of expansion terms using semantic and user interaction features. In: *CIKM 2013*, pp. 1825–1828 (2013)
6. Balaneshin-kordan, S., Kotov, A.: An empirical comparison of term association and knowledge graphs for query expansion. In: *ECIR*, pp. 761–767 (2016)
7. Kotov, A., Zhai, C.: Interactive sense feedback for difficult queries. In: *CIKM*, pp. 163–172 (2011)
8. Kotov, A., Zhai, C.: Tapping into knowledge base for concept feedback: leveraging conceptnet to improve search results for difficult queries. In: *WSDM*, pp. 403–412 (2012)
9. Dalton, J., Dietz, L., Allan, J.: Entity query feature expansion using knowledge base links. In: *SIGIR*, pp. 365–374 (2014)
10. Xiong, C., Callan, J.: Esdrank: Connecting query and documents through external semistructured data. In: *CIKM*, vol. 6, 3–1 (2015)
11. Xiong, C., Callan, J.: Query expansion with freebase. In: *ICTIR*, pp. 111–120 (2015)
12. Balaneshin-kordan, S., Kotov, A.: Sequential query expansion using concept graph. In: *CKIM 2016*, pp. 155–164 (2016)
13. Park, J.-H., Chung, C.-W.: Semantic annotation for dynamic web environment. In: *WWW*, pp. 353–354 (2014)
14. Staykova, K., Agre, G.: Use of ontology-to-text relation for creating semantic annotation. In: *CompSysTech*, pp. 64–71 (2012)
15. Chen, R.-C., Spina, D., Croft, W.B.: Harnessing semantic for answer sentence retrieval. In: *ESAIR 2015*, pp. 21–27 (2015)
16. Rao, B.M., Nanaji, U., Swapna, Y.: Multi-viewpoint based similarity measure and optimality criteria for document clustering. *Adv. Res. Comput. Sci. Softw. Eng.* **2**(6), 232–236 (2012)
17. Kershnerbaum, A., Slyke, R.V.: Computing minimum spanning trees efficiently. In: *ACM annual conference*, pp. 518–527 (1972)





# Improving the Stability for Spiking Neural Networks Using Anti-noise Learning Rule

Yuling Luo<sup>1</sup>, Qiang Fu<sup>1</sup>, Junxiu Liu<sup>1(✉)</sup>, Yongchuang Huang<sup>1</sup>,  
Xuemei Ding<sup>2</sup>, and Yi Cao<sup>3</sup>

<sup>1</sup> Faculty of Electronic Engineering, Guangxi Normal University, Guilin 541004, China  
liujunxiu@mailbox.gxnu.edu.cn

<sup>2</sup> School of Computing, Engineering and Intelligent Systems, Ulster University,  
Derry BT48 7JL, UK

<sup>3</sup> Department of Business Transformation and Sustainable Enterprise, Surrey Business School,  
University of Surrey, Surrey GU2 7XH, UK

**Abstract.** Most of the existing SNNs only consider training the noise-free data. However, noise extensively exists in actual SNNs. The stability of networks is affected by noise perturbation during the training period. Therefore, one research challenge is to improve the stability and produce reliable outputs under the present of noises. In this paper, the training method and the exponential method are employed to enhance the neural network ability of noise tolerance. The comparison of conventional and anti-noise SNNs under various tasks shows that the anti-noise SNN can significantly improve the noise tolerance capability.

**Keywords:** Spiking neural network · Stability · Noise tolerance · Learning rule

## 1 Introduction

The spiking neural networks (SNNs) are getting more and more attention in recent years. A major drive behind this trend is because SNNs have more computational power than traditional artificial neural networks, such as multi-layer perceptron [1], radical basis functions [2], back propagation neural network [3] and so on. Spiking neurons, which are the core component of a SNN, have a descriptive mathematical model of a biological neuron. Therefore, the information that SNN processes is not real input values, but a spike train mimicking the neurons in the brain.

The SNNs have been used for many research and applications, e.g. memory model [4], neuroprosthetics [5], neuromorphic circuits [6], fault-tolerant computing [7], and many other applications [8, 9]. For most applications of SNNs, its stability is of critical importance [10]. However, if the input signal, learning rate or time delay of the network are disturbed, then the network outputs would be affected to a certain extent. Therefore, to improve the stability of SNN is a research challenge. In the traditional artificial neural network, the model parameters, variables are considered to solve this problem. For instance, the approaches of [11, 12] studied the state estimation of neural networks by disturbing the dependent conditions of delay distributions. In the approach of [13], by

analyzing the ways of noise disturbing the neural response, a robust Tempotron (R-Tempotron) learning rule was proposed for spike-timing based decisions. Different to these approaches, this paper focuses on the SNNs with noisy inputs and investigates the strategies to improve the noise tolerance capability of the networks. Firstly, the stability of the network is analyzed under the simulated noisy inputs, i.e. the random perturbation of the input signal in a certain range. Then the learning rule is extended to improve the capacity of resisting disturbances.

The rest of this paper is organized as follows. Section 2 discusses the spike neuron models and learning algorithm briefly. The methods of improving SNN stability is derived in Sect. 3. Simulation experiment results are provided in Sect. 4 and Sect. 5 concludes the paper.

## 2 The SRM Neuron Model and Learning Rule

The SRM neuron model [14] can give a good approximation of synaptic response for the neuron. Therefore, it is used as the target neuron model for the investigation of anti-noise learning rule in this work.

The state of the post-synaptic neuron  $j$  is described by a single variable  $u_j$  in the framework of the SRM. The neuron is at its resting value when  $u_j$  equals to zero if there is no input spikes and it can be denoted by  $u_{rest} = 0$ . If  $u_j$  reaches the threshold after summing the effects of several incoming spikes, an output spike is triggered. After firing, the evolution of  $u_j$  is described by

$$u_j(t) = \eta \left( t - t_j^{(f)} \right) \sum_i \sum_{t_i^{(g)}} \sum_{k=1}^l w_{ji}^k \varepsilon \left( t - t_i^{(g)} - d^k \right), \quad (1)$$

where  $w_{ji}^k$  is the synapse weight from pre-synaptic neuron  $i$  to post-synaptic neuron  $j$  with a delay of  $d^k$ . The  $t_j^{(f)}$  and  $t_i^{(g)}$  denote the spike-times of neuron  $j$  and neuron  $i$ , respectively. The function  $\varepsilon$  describes the time course of the response to an incoming spike, which is given by

$$\varepsilon(s) = \frac{s}{\tau} \exp \left( 1 - \frac{s}{\tau} \right) H(s), \quad (2)$$

where  $\tau$  is the post synaptic membrane potential time constant.  $H(s)$  is the Heavy-side step function. The return of the membrane potential to baseline after an action potential is described by a function  $\eta$ , which is named refractoriness. The refractoriness is characterized experimentally by the observation immediately after a first action potential. It is impossible (absolute refractoriness) or more difficult (relative refractoriness) to excite a second spike.

The Multi-SpikeProp learning rule proposed in [15] is employed for analysis. The error measurement of the network is represented by the sum of the squared differences of the desired and actual spike times, which can be described by

$$E = \frac{1}{2} \sum_{j \in J} \left( t_j^1 - t_d^1 \right)^2, \quad (3)$$

where  $t_d^1$  denotes the desired first spike time of neuron  $j$  and  $J$  denotes the group of neurons in the output layer.

In order to minimize the error of network, the synapse weight from neuron  $i$  to  $j$  should be trained. The weight change function is given by

$$\Delta w_{ji}^k = -\eta \frac{\partial E}{\partial w_{ji}^k}, \quad (4)$$

where  $\eta$  is a constant named by the learning rate. More details of the Multi-SpikeProp learning rule can be found in the approaches of [15–18].

### 3 The Anti-noise SNN Learning Rule

In this section, a solution is proposed to solve the problem of the stability of the network during training period. Two methods are proposed to improve the SNN anti-noise capability for Multi-SpikeProp learning rule.

The method of training time constant of membrane potential is similar as the algorithm proposed in the approach of [19]. The synaptic time constant adopts gradient descent algorithm to adjust it to a suitable value. The error function to the time constant of membrane potential is given by

$$\frac{\partial E}{\partial \tau_{ji}^k} = \frac{\partial E}{\partial t_j} \frac{\partial t_j}{\partial u_j(t)} \frac{\partial u_j(t)}{\partial \tau_{ji}^k}. \quad (5)$$

The calculations of first and second terms (i.e.  $\frac{\partial E}{\partial t_j}$  and  $\frac{\partial t_j}{\partial u_j(t)}$ ) are the same as calculating the synapse weights. The last term  $\frac{\partial u_j(t)}{\partial \tau_{ji}^k}$  can be calculated by

$$\frac{\partial u_j(t)}{\partial \tau_{ji}^k} = w_{ji}^k \frac{\epsilon_{ji}^k}{\partial \tau_{ji}^k} \left( t - t_i^{(g)} - d^k \right) = w_{ji}^k \epsilon_{ji}^k \left( t - t_i^{(g)} - d^k \right) \left[ \frac{\left( t - t_i^{(g)} - d^k \right)}{\left( \tau_{ji}^k \right)^2} - \frac{1}{\tau_{ji}^k} \right]. \quad (6)$$

After calculating the above formulas, the value of  $\frac{\partial E}{\partial \tau_{ji}^k}$  is given by

$$\frac{\partial E}{\partial \tau_{ji}^k} = w_{ji}^k \epsilon_{ji}^k \left( t - t_i^{(g)} - d^k \right) \left[ \frac{\left( t - t_i^{(g)} - d^k \right)}{\left( \tau_{ji}^k \right)^2} - \frac{1}{\tau_{ji}^k} \right] * \frac{-(t_j - t_d)}{\sum_i \sum_k w_{ji}^k \frac{\partial \epsilon_{ji}^k}{\partial t} \left( t - t_i^{(g)} - d^k \right)}. \quad (7)$$

The update function is given by

$$\Delta\tau_{ji}^k = -\eta_\tau \frac{\partial E}{\partial \tau_{ji}^k}, \quad (8)$$

where  $\eta_\tau$  is the learning rate for time constant of membrane potential. It can control the rate of gradient descent during the training period. This method can be used to improve the anti-noise capability of SNN and the results will be given in Sect. 4.

The exponential method is employed to design the anti-noise SNN learning rule in a simple way. The error function ( $E$ ) is employed to adjust the synaptic time constant instead of calculating the gradient. The method of adjusting membrane potential time constant exponentially can be given by

$$\tau_{ji}^{k+1} = \tau_0 e^{aE}, \quad (9)$$

where  $\tau_0$  is initial value of synaptic time constant. The parameter  $a$  controls the value of synaptic time constant. The time constant of the membrane potential changes with the variation of the error function. Once the error function limits to 0,  $\tau_{ji}^{k+1}$  limits to  $\tau_0$ . The synaptic time constant is two times faster if multiply  $\tau_0$  by a number that is greater than one. This is more concise than the method of training time constant of membrane potential.

These two proposed methods achieve the target of anti-noise by adjusting the synaptic time constant, because the synaptic time constant can affect the response degree of post synaptic neurons and the time constants of membrane potentials are different for different neurons in the network. Unlike the Multi-SpikeProp algorithm, the synaptic time constant is not fixed in the anti-noise SNN learning rule which can enhance the noise tolerance capability of the network.

## 4 Experimental Results

The exclusive-or (XOR) [14] and Wisconsin Breast Cancer classification (WBC) [18] task are employed to test the performance of the anti-noise Multi-SpikeProp learning rule in this work. The temporal encoding scheme is chosen for encoding the data. In order to analyze the stability of the anti-noise Multi-SpikeProp learning rule, a random perturbation is designed to add to the input signal. The perturbation method is given by

$$y = x + \sigma, \quad (10)$$

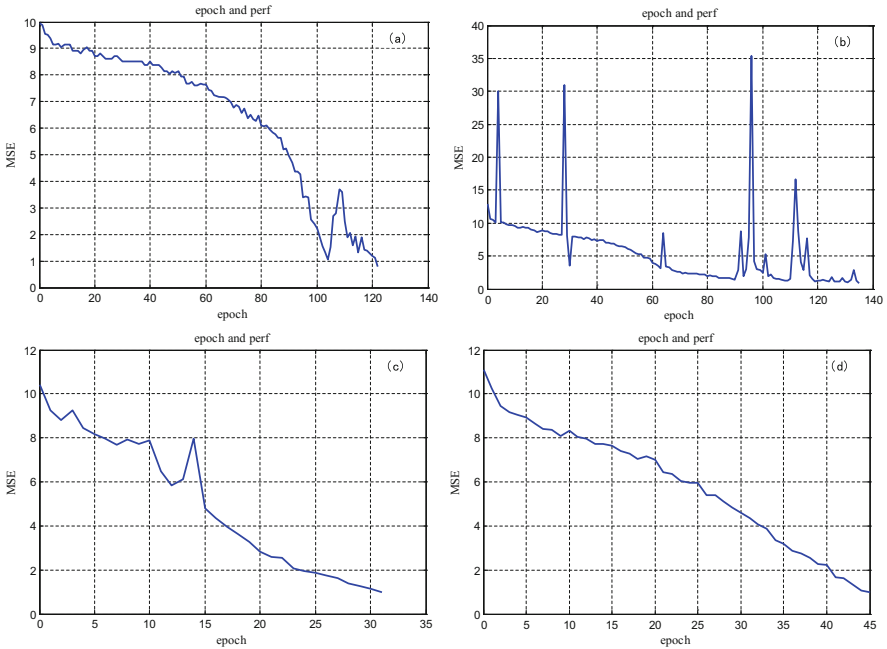
where  $x$  is raw signal,  $\sigma$  is a random perturbation and  $\sigma = 0.1 * \sin(2\pi * b)$  in the experiments where the range of the parameter  $b$  is (0, 1).

### 4.1 XOR Task

For the input signals, an input spike at 0 ms represents logic 0 while a spike at 6 ms represents logic 1. For the output, a spike at 16 ms represents logic 0 while a spike at

10 ms represents logic 1. The time constant of membrane potential should be slightly larger than interval of encoding. Therefore the initial value of  $\tau$  is set to 7 ms. A three-layer feedback network architecture is used in XOR task. As it's not reasonable to add the disturbance to the same pattern, different encoding patterns, e.g. the pattern (0, 6) or (6, 0), are chosen to add the random perturbations.

Figure 1 shows the performance comparison of the different methods. From the process of error reduction during the network training, it can be seen that the original Multi-SpikeProp learning rule [15] exists the concussion in the later period of training, see Fig. 1(a). Figure 1(b) shows the error reduction process using the same learning rule after adding the random perturbations, which is significantly unstable. The training process using the method of training synaptic membrane potential is shown by Fig. 1(c). It can protect against the effect of random disturbance and has less concussion in the latter time period of training than the original Multi-SpikeProp learning rule. Figure 1(d) shows the training process using the method of adjusting the synaptic membrane potential exponentially, which can also counteract the effect of random disturbance and is more stable than the method of training synaptic membrane potential.

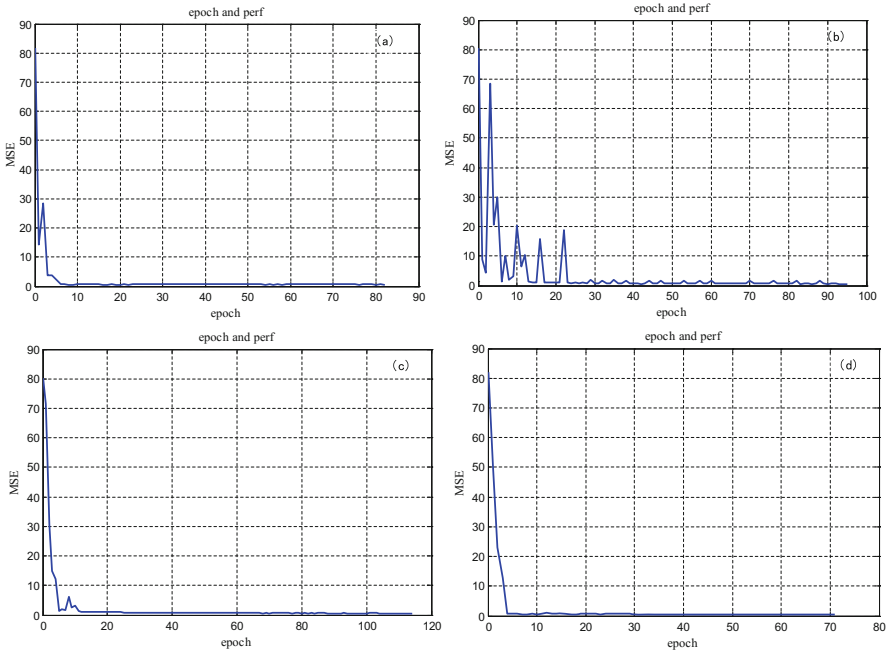


**Fig. 1.** The performance comparison of XOR task using different methods. (a). Original Multi-SpikeProp learning rule. (b). Multi-SpikeProp learning rule with random perturbations. (c). The training process using the method of training synaptic membrane potential. (d). The training process using the method of adjusting the synaptic membrane potential exponentially.

## 4.2 WBC Task

The WBC dataset contains two classes, i.e. benign and malignant cancer tumors, where each sample has 9 attributes. The attributes measure different features of the cytology with integer values in the range 1–10. Each value can be mapped directly to a linear spike train in the range of 20–29 ms. The initial value of  $\tau$  is set to 2 ms. A three-layer feedback network architecture similar to XOR is used in the WBC task. It is consisted of 9, 19, 1 neurons in the input, hidden and output layers, respectively. In the training dataset, 30% of the data are chosen randomly to add the random perturbations.

Figure 2 shows the performance comparison of WBC task using the different methods. From the process of error reduction during the network training, it can be seen that the Multi-SpikeProp learning rule [15] exists the concussion in the early training period, see Fig. 2(a). Figure 2(b) shows the process of error reduction using the same learning rule after adding the random perturbations, which is particularly unstable in the whole training period. The training using the method of synaptic membrane potential is shown by Fig. 2(c). It can protect against the effect of random disturbance and has less concussion in the period of training than the original Multi-SpikeProp learning rule. Figure 2(d) shows the training process using the method of adjusting the synaptic



**Fig. 2.** The performance comparison of WBC task using different methods. (a). Original Multi-SpikeProp learning rule. (b). Multi-SpikeProp learning rule with random perturbations. (c). The training process using the method of training synaptic membrane potential. (d). The training process using the method of adjusting the synaptic membrane potential exponentially.

membrane potential exponentially, which can also counteract the effect of random disturbance and is more stable than the method of training synaptic membrane potential.

Table 1 shows the results of epochs and classification accuracies using Multi-SpikeProp learning rule with/without random perturbation, and two methods of anti-noise SNN learning rule with random perturbation. It can be seen that the highest classification accuracy is given by the method of adjusting the synaptic time constant exponentially.

**Table 1.** The classification accuracies using different learning methods

Learning rule	Epochs	Accuracy rate
Multi-SpikeProp	82	79%
Multi-SpikeProp with random perturbation	95	52%
Training the synaptic time constant	114	77%
Adjusting the synaptic time constant exponentially	71	81%

The Multi-SpikeProp carries out 82 epochs and the classification accuracy is 79% for the dataset without random perturbation. It takes 82 epochs for the dataset with random perturbation and the classification accuracy is very low (52%). The method of training the synaptic time constant carries out 114 epochs and the accuracy rate is 77% for the dataset with random perturbation. The method of adjusting the synaptic time constant exponentially carries out less epochs (71) and the accuracy rate is higher than others (81%).

## 5 Conclusions

The anti-noise SNN learning rule is proposed to improve the anti-noise capability of spiking neural network, which mainly improves the stability using two methods, i.e. the methods of training synaptic membrane potential and adjusting the synaptic time constant exponentially. The training data with a random perturbation is considered in this paper. The experiments of XOR task and WBC classification task are used to verify performance of the proposed anti-noise SNN learning rule. The experiment results showed that the proposed two methods can improve the stability and enhance the noise tolerance capability of the SNNs, and the method of adjusting the synaptic time constant exponentially is more suitable than the method of training synaptic membrane potential.

**Acknowledgement.** This research was supported by the National Natural Science Foundation of China under Grant 61603104, the Guangxi Natural Science Foundation under Grants 2016GXNSFCA380017, 2015GXNSFBA139256 and 2017GXNSFAA198180, the funding of Overseas 100 Talents Program of Guangxi Higher Education, and the Doctoral Research Foundation of Guangxi Normal University under Grant 2016BQ005.

## References

1. Charlotte, Y.-F.H., Bingo, W.-K.L.: Design of multi-layer perceptrons via joint filled function and genetic algorithm approach for video forensics. In: IEEE International Conference on Consumer Electronics-China (ICCE-China), pp. 1–4 (2016)
2. Fei, Y., Hu, J., Gao, K., Tu, J., Li, W., Wang, W.: Predicting risk for portal vein thrombosis in acute pancreatitis patients: A comparison of radical basis function artificial neural network and logistic regression models. *J. Crit. Care* **39**, 115–123 (2017)
3. Wang, J., Wen, Y., Ye, Z., Jian, L., Chen, H.: Convergence analysis of BP neural networks via sparse response. *Appl. Soft. Comput.* **61**, 354–363 (2017)
4. Fumihiro, K., Kato, S., Nakamura, M.: Multi-task reinforcement learning with associative memory models considering the multiple distributions of MDPs. In: Global Conference on Consumer Electronics (GCCE), pp. 27–29 (2015)
5. Blessen, C.E., Douglas, P.M., David, X.C.: Neuroprosthetics in amputee and brain injury rehabilitation. *Exp. Neurol.* **287**, 479–485 (2017)
6. James, C.D., et al.: A historical survey of algorithms and hardware architectures for neural-inspired and neuromorphic computing applications. *Biol. Inspir. Cognit. Archit.* **19**, 49–64 (2017)
7. Liu, J., Harkin, J., Maguire, L.P., McDaid, L., Wade, J.: SPANNER: A self-repairing spiking neural network hardware architecture. In: IEEE Transactions on Neural Networks and Learning Systems, pp. 1–14 (2017)
8. Liu, J., Harkin, J., Mcelholm, M., McDaid, L.: Case study : Bio-inspired self-adaptive strategy for spike-based PID controller. In: IEEE International Symposium on Circuits and Systems, pp. 2700–2703 (2015)
9. Liu, J., Harkin, J., Mcdaid, L., Halliday, D.M., Tyrrell, A.M., Timmis, J.: Self-repairing mobile robotic car using astrocyte-neuron networks. In: International Joint Conference on Neural Networks, pp. 1379–1386 (2016)
10. Shrestha, S.B., Song, Q.: Robust spike-train learning in spike-event based weight update. *Neural Netw* **96**, 33–46 (2017)
11. Yang, J., Yang, W., Wu, W.: A remark on the error-backpropagation learning algorithm for spiking. *Appl. Math. Lett.* **25**(8), 1118–1120 (2012)
12. Bao, H., Cao, J.: Delay-distribution-dependent state estimation for discrete-time stochastic neural networks with random delay. *Neural Netw.* **24**(1), 19–28 (2011)
13. Zhang, M., Li, J., Wang, Y., Gao, G.: R-tempotron: A robust tempotron learning rule for spike timing-based decisions. In: International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), pp. 139–142 (2016)
14. Kawanishi, K., Takase, H., Kawanaka, H., Tsuruoka, S.: Reduce the computing time for SpikeProp by approximation of spike response function. In: 20th International Conference on Knowledge Based and Intelligent Information and Engineering Systems, vol. 96, pp. 1186–1192 (2016)
15. Booi, O., Tat Nguyen, H.: A gradient descent rule for spiking neurons emitting multiple spikes. *Inf. Process. Lett.* **95**(6), 552–558 (2005)
16. Bohte, S.M., Kok, J.N., La Poutre, H.: Error-backpropagation in temporally encoded networks of spiking neurons. *Neurocomputing* **48**(1–4), 17–37 (2002)
17. Luo, Y., Fu, Q., Liu, J., Harkin, J.: An extended algorithm using adaptation of momentum and learning rate for spiking neurons emitting multiple spikes. *Int. Work-Conf. Artif. Neural Netw. (IWANN)* **237**, 569–579 (2017)



18. Fu, Q., et al.: Improving learning algorithm performance for spiking neural networks. In: IEEE 17th International Conference on Communication Technology (ICCT), pp. 1916–1919 (2017)
19. Schrauwen, B., Van Campenhout, J.: Extending spikeprop. In: International Joint Conference on Neural Networks, pp. 471–476 (2004)



# An Improved Convolutional Neural Network Model with Adversarial Net for Multi-label Image Classification

Tao Zhou, Zhixin Li<sup>(✉)</sup>, Canlong Zhang, and Lan Lin

Guangxi Key Lab of Multi-source Information Mining and Security,  
Guangxi Normal University, Guilin 541004, China  
lizz@gxnu.edu.cn

**Abstract.** Convolution neural network (CNN) achieves outstanding results in single-label image classification task. However, due to the complex underlying object layout and insufficient multi-label training images, it is still an open problem that how CNN better handle multi-label images. In this work, we proposes an improved deep CNN model with Adversarial Net which can extract features of objects at different scales in multi-label images by spatial pyramid pooling. In model training, we first transfer the parameters pre-trained on ImageNet to our model, then train an Adversarial Network to generates examples with occlusions and combine it with our model, which make our model invariant to occlusions. Experimental results on Pascal VOC 2012 multi-label image dataset demonstrate the superiority of the proposed approach over many state-of-the-arts approaches.

## 1 Introduction

Deep learning in the multi-classification problem has achieved good results currently. Many various multi-label image classification models have been conducted in recent years. These models are generally based on two types of frameworks, bag-of-words (BoW) [1–4] and deep learning [5, 6]. In recent works, deep learning in the multi-classification problem has achieved good results currently, However, in multi-label images, and the location, shape and scale of each object, are not the same. Moreover, most objects in multi-label images have occlusion, which is also a great challenge. Furthermore, due to the tremendous parameters to be learned for CNN, it requires a lot of images to train CNN model. In addition, the burden of collection and annotation for a large scale multi-label image dataset is generally extremely high. HCP [7], proposes a flexible deep CNN Infrastructure which take full advantage of CNN for multi-label image classification, which achieves good results of 84.2%.

To address these issues, we proposes an improved deep CNN model. We use transfer learning to solve the problem of insufficient multi-label training data, and use SPP layer to solve the problem that the scales of objects are different in multi-label data. Finally, we use the adversarial network to solve the occlusion

problem. The experimental results on two multi-label image datasets, Pascal VOC 2012 and Corel 5K, show that our approach performs more effectively and accurately.

## 2 Improved CNN Model

In our improved model, we transfer the parameters pre-trained on ImageNet to our model and improve the structure of the model by replacing the last pooling layer with a spatial pyramid pooling layer. And we consider sigmoid cross entropy loss as the multi-label loss function. Finally, we train an Adversarial Network to generate occlusion examples. The improved CNN model is based on the VGG16 [8] network structure. It contains 13 convolution layers and contains tens of millions of parameters. Fortunately, a large single-label image dataset, ImageNet, can be used to pre-train the shared CNN structure for parameter initialization. As shown in Fig. 1, the parameters of the VGG16 are transferred to our model. By the parameters transferring, we can reduce the training time, and also achieve a good result.

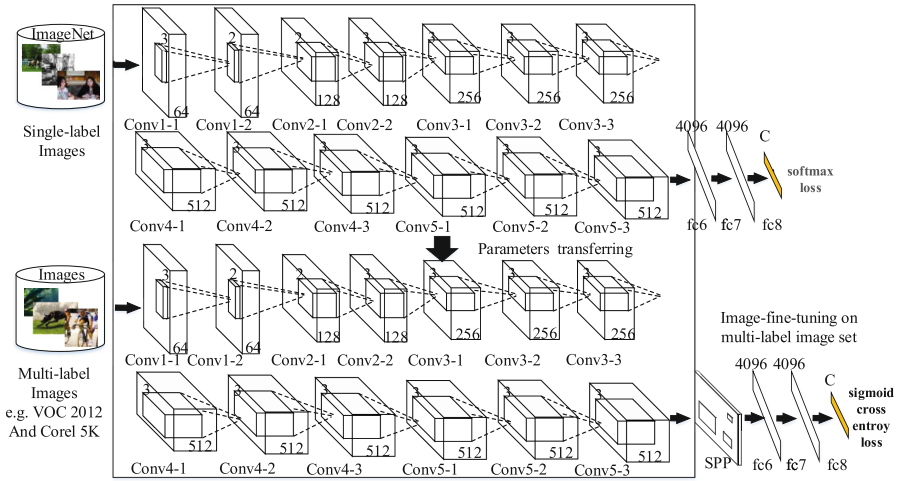
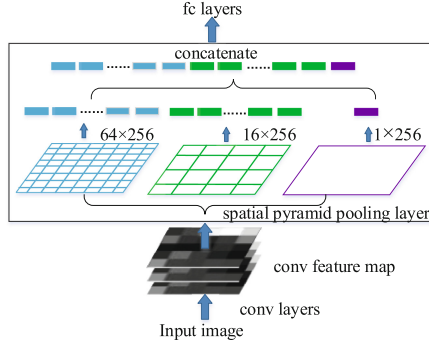


Fig. 1. The improved CNN Model with Parameter transferring.

To enhance the robustness of recognizing objects at different scales, we replace the last pooling layer with a spatial pyramid pooling (SPP) [5] layer. As shown in Fig. 2, we design a three-level pyramid pooling with three different sizes' ( $1 \times 1$ ,  $4 \times 4$ ,  $8 \times 8$ ) bins so that we could get  $64 + 16 + 1 = 81$  bins for a feature map generated by conv5-3 layer. In each spatial bin, we pool the responses of each filter with max pooling. The outputs of the spatial pyramid pooling are  $81 \times 256$ -dimensional vectors (the number of filters in the



**Fig. 2.** The network structure with spatial pyramid pooling layer.

last convolutional layer is 256). The fixed-dimensional vectors are the input to the fully-connected layer.

With spatial pyramid pooling, the input image can be of any size, so that we can train our CNN with multi-scale images. And it can also pool features extracted at variable scales which improves the accuracy of multi-label image classification. We propose using sigmoid cross entropy loss as the multi-label loss function for backward propagation to get more accurate multi-label output. The function can be formulated as,

$$J(\theta) = -\frac{1}{n} \sum_{i=1}^n \sum_{j=1}^c y_i^j \ln(h(x_i)) + (1 - y_i^j) \ln(1 - h(x_i)) \quad (1)$$

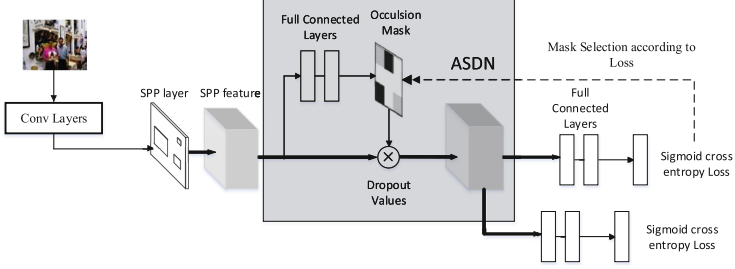
$$h(x) = \frac{1}{1 + e^{-x}} \quad (2)$$

where the  $c$  represents the total number of labels, which is 20 for VOC 2012 dataset,  $(x_i, y_i^j)$  represents the  $i$ -th group of data and its mark corresponding to the  $j$ -th label (0 or 1),  $h(x)$  represents the sigmoid function,  $y_i^j$  takes 0 or 1 to judge whether the  $j$ -th label exists. Cross entropy loss can not only predict the similarity between the predicted label and the real label, but also speed up updating the parameters.

### 3 Using Adversarial Network to Improve Accuracy

Our goal is to make our CNN model robust to occlusion. Fortunately, the author of A Fast RCNN [9] take an alternative approach that use Adversarial Spatial Dropout Network (ASDN) to generate examples of occlusion to train network. As is shown in Fig. 3, the adversarial network shares the convolutional layers and SPP layer with our network. The parameters are not shared in our networks as the two networks are optimized to do the exact opposite tasks. The adversarial network has to learn how to predict the feature on which our network would

fail. We train this adversarial network via the loss function exactly the opposite of Eq. 1. When the feature generated by the adversarial network is easy for our network to classify, we get a high loss for the adversarial network.



**Fig. 3.** Adversarial network architecture, occlusion masks are created to generate hard examples for training.

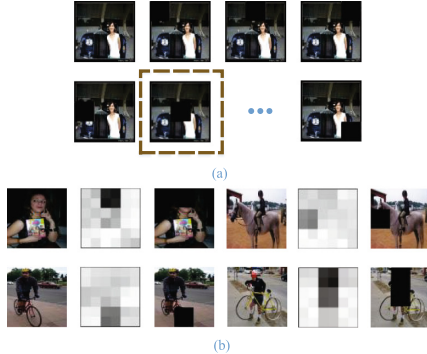
### 3.1 Adversarial Spatial Dropout Network Training

In our experiment, we pre-train the ASDN for creating occlusions before using it to improve our network. After training our network on multi-label image set, we train the ASDN model for creating the occlusions by fixing all the layers in our network. Note that the network ASDN is learned in conjunction with our network during training. As shown in Fig. 3, the convolutional features for each feature map after the SPP layer are obtained as the inputs for the adversarial network. Given a feature map, the ASDN will try to generate a mask indicating which parts of the feature to dropout (assigning zeros) so that the classification of the network will be harder.

The specific process is as follows, given a feature map with size of  $d \times d$  and the sliding window with size of  $d/3 \times d/3$  is applied. The sliding window process is represent by mapping the window back to the image as Fig. 4(a). When the sliding window slides, it overrides the position of the space and deletes the values in all the channels of the corresponding window, where the deleted area generates a new feature vector. Based on all the missing  $d/3 \times d/3$  windows, it passed all the new feature vectors obtained above to the sigmoid cross entropy loss layer to calculate the loss and selected the highest one. Then the window create a single  $d \times d$  mask (with 1 for the window location and 0 for the other pixels) for it. In this way, it generate these spatial masks for  $n$  feature maps and get  $n$  pairs of training samples so that ASDN can generate masks that have high losses. The binary cross entropy loss is used to train ASDN, the formula is as follows,

$$L = -\frac{1}{n} \sum_p^n \sum_{i,j}^d [\tilde{M}_{ij}^p A_{ij}(X^p) + (1 - \tilde{M}_{ij}^p)(1 - A_{ij}(X^p))] \quad (3)$$

where  $A_{ij}(X^p)$  represents the outputs of the ASDN in location  $(i, j)$  given input feature map  $X^p$ . The output generated by ASDN is not a binary mask but rather a continuous heatmap. The ASDN use importance sampling to select the top 1/3 pixels to mask out. More specifically, given a heatmap, 1/3 pixels out of them are selected to assign the value 1 and the rest of 2/3 pixels are set to 0. The network starts to recognize which part of the objects are significant for classification is showed in Fig. 4(b). In this case, we use the masks to occlude parts to make the classification harder.



**Fig. 4.** (a) Examples of occlusions that are sifted to select the hard occlusions (b) Examples of occlusion masks generated by ASDN network.

### 3.2 Joint Learning

Given the pre-trained ASDN in Adversarial Network and our CNN model, we jointly optimize these two networks. For training the CNN model, we first use the ASDN to generate the masks on the features after the SPP-layer during the forward propagation, the ASDN generates binary masks and use them to drop out the values in the features, then forward the modified features to calculate losses and train the CNN model. Note that although our features are modified, the labels remain the same. In this way “harder” and more diverse examples are created for training the CNN model. For training the ASDN in Adversarial Network, since the sampling strategy is applied to convert the heatmap into a binary mask, which cannot directly back-prop the gradients from the classification loss only those hard example masks are used as ground-truth to train the adversarial network by using the same loss as described in Eq. 3 to compute which binary masks lead to significant drops in CNN classification scores.

## 4 Experiment

### 4.1 Datasets and Evaluation Measures

We evaluate the proposed approach on Corel 5K [10] and PASCAL VOC 2012 datasets, which are widely used for classification. These two datasets, which contain 22,531 and 5,000 images respectively, are divided into train, val and test subsets. We conduct our experiments on the trainval/test splits (4,500/500 for Corel 5K 11,540/10,991 for VOC 2012). The evaluation metric is Precision, Recall, Average Precision (AP) and mean of AP (mAP) complying with the PASCAL challenge protocols [11].

### 4.2 Image-Fine-Tuning on Multi-label Image Set

Since we conducted multi-scale training [12] on the dataset, each image is resize into  $256 \times 256$  and  $384 \times 384$ . The fully connected layers fc6 and fc7 are initialized from zero-mean Gaussian distributions with standard deviations 0.01. The first few convolutional layers mainly extract some low-level invariant representations, thus the parameters are quite consistent from ImageNet to the multi-label dataset. So the learning rate of them is set to 0.001. The fully-connected layers are adapted to the new target dataset, so a higher learning rate 0.01 is set to them, and we decrease the learning rates to one tenth of the current ones after every 20 epoches (60 epoches in all).

After being trained on multi-label image set, our network has a sense of the objects in the dataset, we train ASDN for 12K iterations. Given the pre-trained ASDN and our CNN model, we train our joint model for 120K iterations.

### 4.3 Results on Corel 5K and VOC 2012

Table 1 reports results of several models on the set of all 260 words which occur in the Corel 5K. We can find that the model with SPP layer obtain significant improvements of 5.4%, considering the impact of multi-scale training, we give the results of the model with SPP layer only using single-size training whose image size is  $256 \times 256$  or  $384 \times 384$ . The results show that the improved network structure is superior to the no-SPP structure which mainly due to the multi-scale training rather than the improved structure itself. This is mainly because the objects occupy smaller regions in VOC 2012 and Corel 5K but larger regions in ImageNet, so training the network with additional image size of  $384 \times 384$  can partially address this “scale mismatch” issue. The accuracy of the model was also improved obviously by ASDN. Table 2 reports the experimental results compared with the state-of-the-art approaches on VOC 2012. It demonstrate the superiority of the proposed approach over other state-of-the-arts. Classifications of images obtained by proposed approach are showed in Fig. 5. It shows the robustness of our approach to occlusions and small object.






**Table 1.** Comparing the classification results with traditional methods.

Approach	Precision	Recall	mAP
PLSA-WORDS	22.1	12.1	19.1
HGDM	32.1	29.3	26.3
OverFeat	37.5	36.5	35.7
I-FT -no SPP	35.9	34.4	34.6
I-FT -single size $256 \times 256$	36.5	35.0	35.1
I-FT -single size $384 \times 384$	39.5	38.1	38.4
I-FT	41.3	39.8	40.0
I-FT with ASDN (Our Approach)	45.2	41.5	43.3

**Table 2.** Image classification results on Pascal VOC 2012, compared to other approaches.

	I-FT	SPP(OFF)	OF	AGS	NUS-PSL	PRE	HCP	Our method
aero	95.4	96.2	93.3	95.9	97.3	93.5	97.5	96.3
bike	83.7	85.3	83.6	83.2	84.2	84.2	84.2	84.2
bird	87.2	88.0	82.3	79.0	80.8	87.7	93.0	90.3
blt	56.4	59.2	54.4	57.5	60.8	57.3	62.5	57.3
boat	85.7	86.5	83.5	84.0	85.3	80.9	89.4	86.1
bus	90.1	93.9	90.7	91.4	89.9	85.0	90.2	90.9
car	85.5	86.3	83.7	84.3	86.8	81.6	84.6	86.2
cat	89.0	86.8	84.8	83.4	89.3	89.4	94.8	94.3
chair	66.8	70.6	67.7	70.2	75.4	66.9	69.7	67.2
cow	76.1	76.9	74.1	75.1	77.8	73.8	90.2	82.0
dog	86.6	79.4	77.7	78.2	83.0	89.5	93.4	91.4
hrs	85.2	86.0	83.8	85.4	87.5	83.2	93.7	94.4
mbk	87.8	90.7	87.5	88.4	90.1	87.6	88.8	88.1
per	94.1	94.9	92.7	92.8	95.0	95.8	93.3	94.5
plant	55.3	56.1	53.4	52.4	57.8	61.4	59.7	64.9
shp	79.9	80.7	77.2	78.5	79.2	79.0	90.3	82.4
sofa	69.5	70.3	68.1	67.8	73.4	54.3	61.8	76.8
table	69.2	70.0	66.9	68.9	75.1	62.0	74.1	74.1
train	94.8	95.6	91.6	93.0	94.5	88.0	94.4	96.6
tv	81.9	82.7	79.4	77.4	80.7	78.3	78.0	82.0
mAP	81.0	81.8	78.8	79.3	82.2	78.7	84.2	84.0



Image					
Ground Truth	bicycle, bottle, person	chair, diningtable, sofa	bottle, chair, person, sofa	bottle, chair, diningtable, person	chair, diningtable, person, sofa, tvmonitor
Overfeat	person, bottle	chair, diningtable	chair, person, sofa	bottle, person	person, tvmonitor, sofa
Our Method	bottle, diningtable, person	bottle, chair, diningtable, sofa	chair, person, sofa, bottle	diningtable, person, bottle	chair, person, sofa, tvmonitor

**Fig. 5.** Comparison of classification made by OverFeat and Our method on VOC2012.

## 5 Conclusion

In this paper, an improved deep CNN model is proposed, the parameters pre-trained on large-scale single-label image dataset, ImageNet, are transferred to tackle the insufficient multi-label problem. And the SPP layer as well as ASDN are used to improve the mAP, which makes the model robust to the objects at different scales and objects with occlusions. In comparison to many state-of-the-art approaches, experimental results show that our approach achieves superior results in multi-label image classification on VOC 2012 and Corel 5K.

**Acknowledgements.** This work is supported by the National Natural Science Foundation of China (Nos. 61663004), the Guangxi Natural Science Foundation (Nos. 2016GXNSFAA380146, 2017GXNSFAA198365), the Research Fund of Guangxi Key Lab of Multi-source Information Mining and Security (16-A-03-02), the Guangxi Special Project of Science and Technology Base and Talents (AD16380008) and the Guangxi Collaborative Innovation Center of Multi-source Information Integration and Intelligent Processing.

## References

- Zan, W., Tsim, Y.C., Yeung, W.S., Chan, K.C.: Probabilistic latent semantic analyses (plsa) in bibliometric analysis for technology forecasting. *J. Technol. Manag. Innov.* **2**(1), 11–24 (2007)
- Li, Z., Shi, Z., Zhao, W., Li, Z., Tang, Z.: Learning semantic concepts from image database with hybrid generative/discriminative approach. *Eng. Appl. Artif. Intell.* **26**(9), 2143–2152 (2013)
- Dong, J., Xia, W., Chen, Q., Feng, J., Huang, Z., Yan, S.: Subcategory-aware object classification. In: *Proceedings of CVPR*, pp. 827–834 (2013)
- Azizpour, H., Razavian, A.S., Sullivan, J., Maki, A., Carlsson, S.: Factors of transferability for a generic convnet representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **38**(9), 1790–1802 (2015)
- He, K., Zhang, X., Ren, S., Sun, J.: Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **37**(9), 1904–1916 (2015)
- Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., Lecun, Y.: Overfeat: Integrated recognition, localization and detection using convolutional networks. In: *ICLR* (2014)

7. Wei, Y., et al.: HCP: a flexible CNN framework for multi-label image classification. *IEEE Trans. Pattern Anal. Mach. Intell.* **38**(9), 1901–1907 (2016)
8. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: *ICLR* (2014)
9. Wang, X., Shrivastava, A., Gupta, A.: A-fast-RCNN: hard positive generation via adversary for object detection. In: *Proceedings of the CVPR*, pp. 21–26 (2017)
10. Duygulu, P., Freitas, N.D., Barnard, K., Forsyth, D.A.: Object recognition as machine translation. In: *Proceedings of the ECCV*, pp. 97–112 (2002)
11. Everingham, M., Gool, L.V., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classes (VOC) challenge. *Int. J. Comput. Vis.* **88**(2), 303–338 (2010)
12. Oquab, M., Bottou, L., Laptev, I., Sivic, J.: Learning and transferring mid-level image representations using convolutional neural networks. In: *Proceedings of the CVPR*, pp. 1717–1724 (2014)



# Integrating Multiscale Contrast Regions for Saliency Detection

Taizhe Tan<sup>1,2</sup>, Qunsheng Zeng<sup>1,2(✉)</sup>, and Kangxi Xuan<sup>1</sup>

<sup>1</sup> Guangdong University of Technology, Panyu District, Guangzhou, China  
1484821795@qq.com, 969313709@qq.com, 1056808552@qq.com

<sup>2</sup> Synergy Innovation Institute of GDUT, Heyuan, China

**Abstract.** Visual saliency detection has lately witnessed substantial progress attributed to powerful feature representation leveraging deep convolutional neural networks (CNNs). However, existing CNN-based method has a lot of redundant computation resulting in inferring saliency maps is very time-consuming. In this paper, we propose a multiscale contrast regions deep learning framework employed to calculate salient score of an integrated image. Experimental results demonstrate that our approach is capable of achieving almost the same performance on the four public benchmarks compared to the relevant method MDF. Meanwhile, the computational efficiency is remarkably improved, when inferring the image of 400 \* 300 size only takes average 3.32 s using our algorithm while MDF method consumes 8.0 s reducing rough 60% cost.

**Keywords:** Saliency detection · CNNs · Redundant computation · Multiscale

## 1 Introduction

Visual saliency detection aims to highlight interesting regions or objects in a picture as possible as consistent with human perception. In particular, it often serves as a pre-processing step for many computer visions and image processing tasks including object detection and recognition, image or video compression [1], image cropping [2], image segmentation [3, 4], video event detection [5] and so on.

Achievements from cognitive research [6, 7] scientifically indicate that image regions contrast is the most important factor in visual attention for saliency detection. Lots of traditional saliency detection algorithms have been successfully presented using local or global contrast [8, 9]. Many types of handcrafted low-level features, such as texture, color, intensity, are often adopted as contrast features at the pixel or segment level. Unfortunately, handcrafted methods have a good performance in specific scenarios but clearly worsen on other public datasets.

Recently CNNs have been extensively employed, performing better when compared with conventional state of the art [11, 12]. Lately CNN models are presented by operating all segment-level superpixels rather than classifying pixels level, called as MDF algorithm [11]. This method explicitly takes lots of cost attributed to redundant computations.

We propose a contrast learning network via integrating multiscale contrast regions to overcome the aforementioned deficiency of redundant computation.

In summary, the contributions of this paper are as follow:

- (1) We propose an effectively computational deep model by integrating multiscale contrast context including image segmented patch and enclosed neighbor regions as well as whole map reducing the redundancy operations related algorithm.
- (2) A brief scheme is designed for fusing multi-level saliency maps: A set of parameters are learned to integrate multiple maps produced from ours CNN into a salient map, and then feed it into a sigmoid function aiming at transforming the binary saliency score into continuous values unlike other methods applying a heavy fully connected CRF to further refine spacial coherence.
- (3) Experimental results on four benchmark datasets and comparisons with the state-of-the-art MDF approaches demonstrate the great superiority in the time consuming, inferring the size of  $400 * 300$  image only takes average 3.32 s while MDF method consumes 8.0 s.

## 2 Related Work

Two groups of saliency detection algorithm can be divided in terms of different views: bottom-up methods, top-down strategies.

Bottom-up methods include local and global contrast. Local approaches [10, 15, 16] primarily focus on the local contrast between each image element (pixel, region, or patch) and its surroundings to design saliency cues. Global methods assign saliency values of each element according to the uniqueness of each image element over the holistic statistics. For example, Cheng et al. [9] calculated color histograms of each region in the image, and then compared the statistical information of the region with other blocks to construct a global contrast. A frequency-tuned strategy in [17] was applied to identify pixel-level saliency detection by taking advantage of features of color and luminance. Perazzi et al. [18] applied the uniqueness of each element and the spatial distribution of the image to measure global contrast so as to derive saliency regions. A multi-layer model, proposed by Yan et al. [19], deal with small-scale high-contrast patterns in an image.

Top-down pattern is usually task-dependent and object-oriented, utilizing high-level prior knowledge to design saliency detection algorithms [21–23]. For example, a top-down object detectors model presented to identify faces, animals, humans and so on, was constructed by Judd et al. [24]. Borji et al. [25] built a multi-objects detector by means of combining bottom-up cues with top-down features with objective prior knowledge. In paper [26], it proposed a top-down saliency model by training the dictionary modulated by Conditional Random Field (CRF). However, it is difficult to suit multi-categories saliency for these methods.

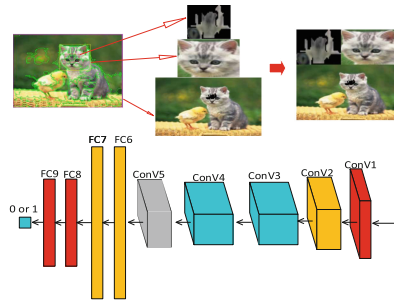
Some researchers had also extensively attempted to employ CNN to detect salient object. For example, Li et al. [11] trained a segment-wise CNN levered to identify saliency values of the multiscale features extracted from AlexNet FC7 layer [13]. In [12], a novel algorithm was proposed to integrate both local cues and global features by

adopting a deep neural network (DNN-L) to learn local patch features and using another deep neural network (DNN-G) to get global information. Similarly, in paper [14], Zhao et al. designed a multi-context deep neural network by considering local contrast and global context aiming to address low-contrast background cases.

### 3 Integrating Multiscale Contrast Regions Deep Network

The flowing pipeline of ours deep learning network mainly composes of two computation components including multiscale contrast regions generated and saliency score calculated.

Motivated by reducing the redundancy computation, our insight originates from the size of segmented regions. We find that, specifically, each patch has single cues and small size, thus it is unnecessary to resize the patch to  $227 * 227$  image scale as MDF. As shown in the above Fig. 1, we exact three rectangles including segmented patch and immediate surrounding regions as well as whole image context aiming at forcing a contrast from enclosed surrounding regions to global context. Meanwhile, we deal with three nested windows before integrating into together, respectively, pixels inside super-pixel rectangle but outside the segmented region and pixels in entire image and inside the superpixel are given mean value calculated over cross training datasets. In the end, a  $227 * 227$  size multiscale contrast image is constructed by integrating three exacted multiscale maps rescaled to different fixed size. The integrated image reserves not only both local cues and global ones but also semantic understanding.



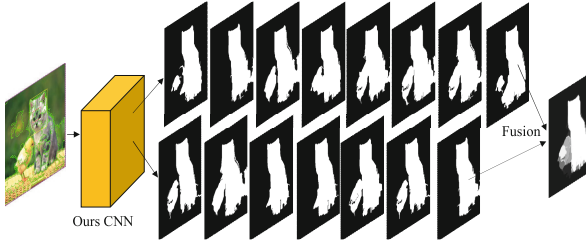
**Fig. 1.** Merging multiscale contrast regions deep CNN pipeline

We leverage AlexNet to play a high-dimension discriminative feature vector extractor role (output FC7 layer 4096 dimensions vector), and additionally two fully connected layers with 300 neurons each layer with one output tier is employed as a higher-level decision maker to compute saliency score [11]. Before training, every training sample is first decomposed into a set of segmented regions whose saliency label is further estimated based on pixel-wise saliency label. Only those superpixels with 70% or more same saliency pixels are qualified as training positive label. Those superpixels with 30% or less same non-saliency pixels are qualified as training negative label.

As a result, our deep saliency model, trained in the MSRA-B benchmark achieves a high performance about 90% test accuracy taking around 16 h. Specifically, learning rate is  $1e-4$ , batch size is 256 and dropout is 0.85.

## 4 Multi-level Maps Fusion

We adopt a multi-level segmentation strategy at various granularity to accommodate as possible as many characters [27], here using 15 groups of parameters  $L_1, L_2, \dots, L_{15}$ . Result in producing 15 saliency maps  $M_1, M_2, \dots, M_{15}$ , as shown in Fig. 2.



**Fig. 2.** Multi-level saliency maps are fused into a refined image

Aiming to further fuse them together to obtain a final optimal saliency map, thus a group of learnable parameters  $a_k$  ( $k \in 1, 2, \dots, 15$ ) is learned as a non-linear combination of the multi-level maps, as shown in Eq. (1).

$$S^* = \frac{1}{1 + e^{-\sum_{k=1}^{15} a_k M_k + b}} \quad (1)$$

Where  $S^*$  is a final optimal saliency map, and the  $a_k$  ( $k \in 1, 2, \dots, 15$ ) parameters is trained by minimizing the follow loss function as shown in Eq. (2).  $N$  is the number of image pixels,  $S_j$  and  $S_j^*$  is respectively the  $j$  pixel of label image and fused saliency map.

$$l_{BCE} = -\frac{1}{N} \sum_{j=1}^N S_j \log(S_j^*) + (1 - S_j) \log(1 - S_j^*) \quad (2)$$

## 5 Experiment Result

Results indicate that our approach can achieve almost the same effect on the four public benchmarks respectively MSRA-B(test) [10], HKU-IS [11], Pascal-s [28], ECSSD [29], saving time cost compared to the relevant method MDF. As below listed in Table 1, our experimental platform is based on a Titan XP and an i7-7700k processor Visual comparison of detecting saliency results is demonstrated as Fig. 3.

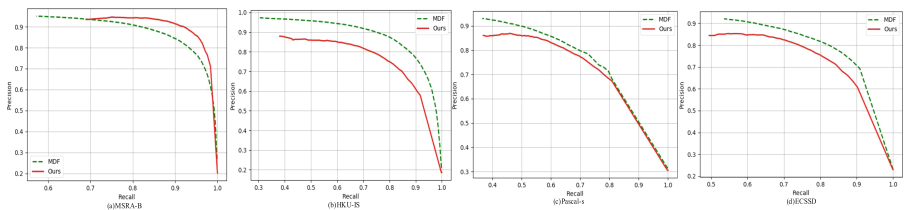
**Table 1.** Time cost comparison inferring saliency regions

	Image size	Deploy device	Code platform	Time cost (Sec)
MDF	400 * 300	GTX Titan Black GPU	MATLAB	8.0
Ours	400 * 300	GTX Titan XP GPU	Pycharm	3.32

**Fig. 3.** Visual comparison of saliency maps

## 5.1 Performance Comparison

A continuous saliency map can be converted into a binary value map by setting a threshold, and a pair of precision-recall (PR) values is calculated when the binary mask is compared against the ground truth. As drawn in Fig. 4., it is clear to conclude that our deep saliency model is capable of achieving almost the same effect on the four public benchmarks compared with MDF in spite of weakness in the HKU-IS while strengthen in the MSRA-B dataset.

**Fig. 4.** Precision-recall curves performance in four public datasets

Moreover, another evaluation criteria, F-measure [20], is applied as the measure of performance. Its computation is as follow Eq. (3).

$$F_{\beta} = \frac{(1 + \beta^2) \cdot Precision \cdot Recall}{\beta^2 \cdot Precision + Recall} \quad (3)$$

$$T_a = \frac{2}{W \times H} \sum_{x=1}^W \sum_{y=1}^H S^*(x, y) \tag{4}$$

Where  $\beta^2$  is equal to 0.3 to adjust precision more important than recall suggested in [27]. *Precision* and *Recall* is computed by using an adaptive threshold  $T_a$  as shown in Eq. (4).  $W$  and  $H$  is the width and height of the saliency map  $S^*$ , and  $S^*(x, y)$  is the pixel value at  $(x, y)$  position of the saliency map.

As shown in Fig. 5 demonstrates that our performance by three evaluation methods, precision and recall as well as F-measure, is almost equal to MDF in MSRA-B, HKU-IS, Pascal-s, ECSSD four public datasets.

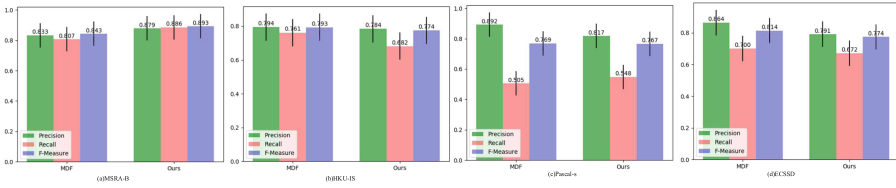


Fig. 5. Precision-recall and F-measure performance using an adaptive threshold

Although precision-recall can reflect the performance of the algorithm, they fail to consider the true negative pixels in the saliency image. A pixelwise criterion, called mean absolute error (MAE), are also employed to evaluate our deep model. The formula is listed as Eq. (5). Where  $S^*(x, y)$  denotes the pixel value at  $(x, y)$  position of the saliency map and  $S(x, y)$  is the pixel  $(x, y)$  value in the label image.

$$MAE = \frac{1}{W \times H} \sum_{x=1}^W \sum_{y=1}^H |S^*(x, y) - S(x, y)| \tag{5}$$

The *MAE* value computed in Fig. 6 is also shown that our methods and MDF have the similar performance in four public datasets.

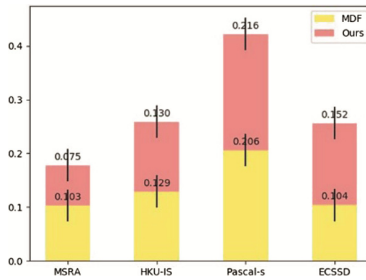


Fig. 6. MAE performance over four public datasets



## 5.2 Speed Improvement

Based on the premise of the same network model, inferring the size of  $400 * 300$  image only takes average 3.32 s using our deep saliency algorithm while MDF method consumes 8.0 s but slight difference in computing device, detailedly as shown in below Table 1. It further proves that our model is capable of reducing redundant computations for saving time cost.

In addition, we also employ our model to test the time cost by inferring larger image sizes, as drawn in Fig. 7, indicating that the increasement of time cost is nearly linearly related with image size.

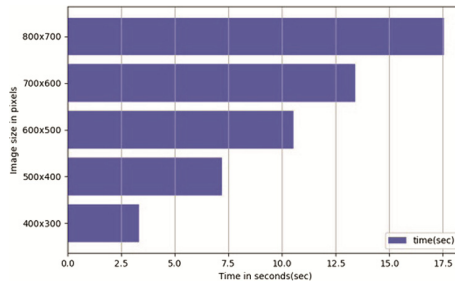


Fig. 7. Time cost in different image size

## 6 Conclusions

In this work, we propose a multiscale contrast regions deep learning framework employed to calculate salient score of an input image integrated with segmented patch and immediate regions as well as global context in order to reduce redundant computations. Moreover, we adopt a multi-level segmentation strategy to accommodate more characters of different images, resulting in 15 saliency maps are produced. Meanwhile, a brief scheme is designed for fusing multi-level saliency maps via nesting linear summation into sigmoid function aiming at transforming the binary saliency score into continuous values. Experiment demonstrate that our method can make sure the performance is nearly close to relevant algorithm MDF reducing time consuming.

## References

1. Cheng, M.M., Warrell, J., Lin, W.Y., Zheng, S., Vineet, V., Crook, N.: Efficient salient region detection with soft image abstraction. In: International Conference on Computer Vision, Sydney, pp. 1529–1536. IEEE (2013)
2. Marchesotti, L., Cifarelli, C., Csurka, G.: A framework for visual saliency detection with applications to image thumbnailing. In: 12th International Conference on Computer Vision, Kyoto, pp. 2232–2239. IEEE (2009)

3. Zou, W., Liu, Z., Kpalma, K., Ronsin, J., Zhao, Y., Komodakis, N.: Unsupervised joint salient region detection and object segmentation. *IEEE Trans. Image Process.* **11**(24), 3858–3873 (2015)
4. Alpert, S., Galun, M., Basri, R., Brandt, A.: Image segmentation by probabilistic bottom-up aggregation and cue integration. In: *Conference on Computer Vision and Pattern Recognition*, Minneapolis, pp. 1–8. IEEE (2007)
5. Huan, W., Guo, H., Wu, X.: Saliency attention based abnormal event detection in video. In: *International Conference on Robotics and Biomimetics (ROBIO 2014)*, Bali, pp. 1039–1043. IEEE (2014)
6. Einhäuser, W., König, P.: Does luminance-contrast contribute to a saliency map for overt visual attention? *Eur. J. Neurosci.* **17**(5), 1089 (2003)
7. Parkhurst, D., Law, K., Niebur, E.: Modeling the role of salience in the allocation of overt visual attention. *Vis. Res.* **42**(1), 107–123 (2002)
8. Yang, C., Zhang, L., Lu, H., Ruan, X., Yang, M.H.: Saliency detection via graph-based manifold ranking. In: *Conference on Computer Vision and Pattern Recognition*, Portland, pp. 3166–3173. IEEE (2013)
9. Cheng, M.M., Zhang, G.X., Mitra, N.J., Huang, X., Hu, S.M.: Global contrast based salient region detection. In: *Conference on Computer Vision and Pattern Recognition*, Providence, pp. 409–416. IEEE (2011)
10. Liu, T., Sun, J., Zheng, N.N., Tang, X., Shum, H.Y.: Learning to detect a salient object. In: *Conference on Computer Vision and Pattern Recognition*, Minneapolis, pp. 1–8. IEEE (2007)
11. Li, G., Yu, Y.: Visual saliency based on multiscale deep features. In: *Computer Vision and Pattern Recognition (CVPR)*, Boston, pp. 5455–5463. IEEE (2015)
12. Wang, L., Lu, H., Ruan, X., Yang, M.H.: Deep networks for saliency detection via local estimation and global search. In: *Computer Vision and Pattern Recognition (CVPR)*, Boston, pp. 3183–3192. IEEE (2015)
13. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: *International Conference on Neural Information Processing Systems*, pp. 1097–1105. Curran Associates Inc. (2012)
14. Zhao, R., Ouyang, W., Li, H., Wang, X.: Saliency detection by multi-context deep learning. In: *Computer Vision and Pattern Recognition (CVPR)*, Boston, pp. 1265–1274. IEEE (2015)
15. Schölkopf, B., Platt, J., Hofmann, T.: Graph-based visual saliency. In: *19th Proceedings of Neural Information Processing Systems*, pp. 545–552. MIT Press (2007)
16. Itti, L., Koch, C., Niebur, E.: A model of saliency-based visual attention for rapid scene analysis. *Trans. Pattern Anal. Mach. Intell.* **11**(20), 1254–1259 (1998)
17. Achanta, R., Hemami, S., Estrada, F., Susstrunk, S.: Frequency-tuned salient region detection. In: *Conference on Computer Vision and Pattern Recognition*, Miami, pp. 1597–1604. IEEE (2009)
18. Perazzi, F., Krähenbühl, P., Pritch, Y., Hornung, A.: Saliency filters: contrast based filtering for salient region detection. In: *Conference on Computer Vision and Pattern Recognition*, Providence, pp. 733–740. IEEE (2012)
19. Yan, Q., Xu, L., Shi, J., Jia, J.: Hierarchical saliency detection. In: *Computer Vision and Pattern Recognition*, Portland, pp. 1155–1162. IEEE (2013)
20. Chen, C., Li, S., Qin, H., Hao, A.: Structure-sensitive saliency detection via multilevel rank analysis in intrinsic feature space. *IEEE Trans. Image Process.* **8**(24), 2303–2316 (2015)
21. Shen, X., Wu, Y.: A unified approach to salient object detection via low rank matrix recovery. In: *Conference on Computer Vision and Pattern Recognition*, Providence, pp. 853–860. IEEE (2012)

22. Jia, Y., Han, M.: Category-independent object-level saliency detection. In: International Conference on Computer Vision, Sydney, pp. 1761–1768. IEEE (2013)
23. Chang, K.-Y., Liu, T.-L., Chen, H.-T., Lai, S.-H.: Fusing generic objectness and visual saliency for salient object detection. In: 14th International Conference on Computer Vision, Barcelona, pp. 914–921. IEEE (2011)
24. Judd, T., Ehinger, K., Durand, F., Torralba, A.: Learning to predict where humans look. In: 12th International Conference on Computer Vision, Kyoto, pp. 2106–2113. IEEE (2009)
25. Borji, A.: Boosting bottom-up and top-down visual features for saliency estimation. In: Conference on Computer Vision and Pattern Recognition, Providence, pp. 438–445. IEEE (2012)
26. Yang, J., Yang, M.H.: Top-down visual saliency via joint CRF and dictionary learning. In: Conference on Computer Vision and Pattern Recognition, Providence, pp. 2296–2303. IEEE (2012)
27. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient graph-based image segmentation. *Int. J. Comput. Vis.* **59**(2), 167–181 (2004)
28. Everingham, M., Williams, C.: The PASCAL Visual Object Classes Challenge 2010 (VOC2010) Part 1 – Challenge & Classification Task. Challenge (2010)
29. Shi, J., Yan, Q., Xu, L., Jia, J.: Hierarchical image saliency detection on extended CSSD. *IEEE Trans. Pattern Anal. Mach. Intell.* **4**(38), 717–729 (2016)



# Automatic Conditional Generation of Personalized Social Media Short Texts

Ziwen Wang<sup>1,2(✉)</sup>, Jie Wang<sup>3</sup>, Haiqian Gu<sup>2,4</sup>, Fei Su<sup>2,4</sup>,  
and Bojin Zhuang<sup>3</sup>

<sup>1</sup> School of Science, Beijing University of Posts and Telecommunications,  
Beijing, China

wangziwen@bupt.edu.cn

<sup>2</sup> Beijing Key Laboratory of Network System and Network Culture,  
Beijing, China

{mixiu, sufeif}@bupt.edu.cn

<sup>3</sup> Ping An Technology (Shenzhen) Co., Ltd., Shenzhen, China

{wangjie388, zhuangbojin232}@pingan.com.cn

<sup>4</sup> School of Information and Communication Engineering,  
Beijing University of Posts and Telecommunications, Beijing, China

**Abstract.** Automatic text generation has received much attention owing to rapid development of deep neural networks. In general, text generation systems based on statistical language model will not consider anthropomorphic characteristics, which results in machine-like generated texts. To fill the gap, we propose a conditional language generation model with Big Five Personality (BFP) feature vectors as input context, which writes human-like short texts. The short text generator consists of a layer of long short memory network (LSTM), where a BFP feature vector is concatenated as one part of input for each cell. To enable supervised training generation model, a text classification model based convolution neural network (CNN) has been used to prepare BFP-tagged Chinese micro-blog corpora. Validated by a BFP linguistic computational model, our generated Chinese short texts exhibit discriminative personality styles, which are also syntactically correct and semantically smooth with appropriate emoticons. With combination of natural language generation with psychological linguistics, our proposed BFP-dependent text generation model can be widely used for individualization in machine translation, image caption, dialogue generation and so on.

**Keywords:** Natural language generation · Deep neural network  
Recurrent neural network · Convolution neural network · Big Five Personality

## 1 Introduction

Natural language generation (NLG) has been intensively studied owing to its important applications such as automatic dialogue generation [1], machine translation [2], text summarization [3], image captions [4] and so on. Based on deep learning approaches, much research effort has been paid to improving the quality of automatically-generated texts in terms of syntax and semantics. For instance, Karpathy (2015) put forward a

character-level recurrent neural network (Char-RNN) for generation tasks [5]. With contextual controlling conditions, sequence to sequence (Seq2Seq) models for machine translation and smart dialogue generation were first introduced by Google Brain in 2014 [6]. Furthermore, Guo (2015) proposed reinforcement learning (RL) scheme which built a deep Q-network as a language generator [7]. Zhang et al. (2016) tried generative adversarial networks (GAN) theory for NLG which employed CNN as the discriminator and LSTM as the generator [8].

Most newly-developed language generation models concentrate on grammatical correction and semantic correlation, however, an essential issue has not been considered: in comparison with texts composed by human beings, machine-generated ones lack of personalities. Psychological linguistics demonstrates that contents and the way people write show their personalities, which could hardly be achieved by NLG robots. Nevertheless, language generation with personalities is demanding in some important application cases, including intelligent customer service and chat bots.

To fill this gap, Li et al. (2016) encoded users' background information in distributed embeddings for neural response generation [9]. On contrast, we encode users' BFP information as contextual control in natural language generation. In this paper, we propose an efficient BFP computation model through classifying social media texts based on a CNN network. By concatenating BFP feature vector as partial input of LSTM cell, a conditional text generation model has been trained which write personality-discriminative short texts.

The rest of this paper is organized as follows. Section 2 presents our conditional language generation model, consisting of a CNN-based BFP classification sub-model and an LSTM-based text generator. Section 3 shows and analyzes generation results and Sect. 4 discusses the conclusions and our limitations.

## 2 The Conditional Language Generation Model

The BFP theory is one of the most important psychological theories for profiling personality of human beings. Amongst it, five big factors have been defined individually as extraversion (E), agreeableness (A), conscientiousness (C), neuroticism (N) and openness (O). As demonstrated by previous reports [10–12], people's BFP traits can be inferred by texts they shared on social media. Based on Linguistic Inquiry and Word Count (LIWC), the BFP of social media users can be analyzed by counting word frequency of certain psychological lexicons. Here, we combine natural language generation with psychological linguistics to propose a BFP-dependent short text generation model. The BFP-based generation model's architecture is shown in Fig. 1. A CNN-based text classification model is trained to achieve BFP representations of micro-blog texts and provide tagged corpora for language generation model training. The main text generator is based on a LSTM language model, which receives a BFP feature vector as the contextual input. With preprocessed Chinese corpora, the generation model is supervise-trained and then used for composing text sequences automatically. Eventually, BFP properties of such generated short texts have been validated based on the psychological linguistic computation method.

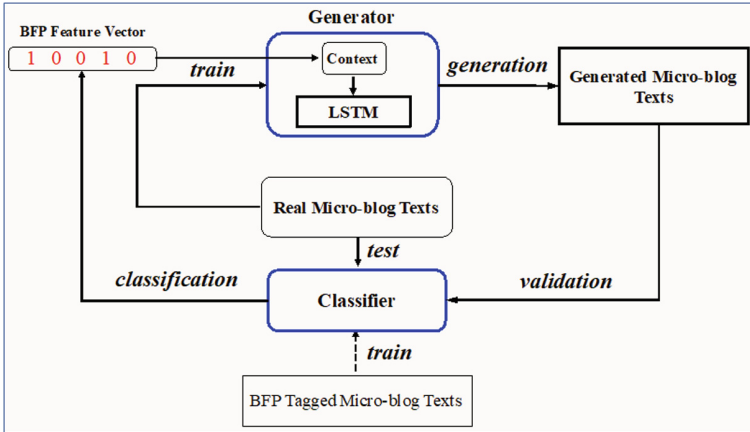


Fig. 1. The architecture of the BFP-dependent conditional generation model.

### 2.1 Psychological Text Classification Model

Lin et al. identified the association between Chinese words and phrases and BFP traits based on SC-LIWC (a simplified Chinese version of LIWC developed by Gao et al. [13]) [12]. Based on SC-LIWC and word-frequency-to-BFP mapping matrix, we have prepared BFP-labelled corpora with the collected social media texts of 17,000 Sina Weibo users (<http://www.weibo.com>) to train a CNN-based Psychological text classification model. After being supervise-trained, the text classification model can be used to collect social media texts into bins of high or low BFP traits.

The text classifier is designed as shown in Fig. 2. Every input Chinese word is embedded into  $k$  dimensional representation vector by word2vec [14]. A convolution operation is then applied with a filter  $w \in \mathbb{R}^{mk}$ , which acts on a window of  $m$  words to abstract a higher level feature, in practice,  $m$  is set as 3 and Relu activation function is applied after the convolution operation. To capture the most important linguistic features, a max-over-time pooling operation is applied in each feature map. Eventually, all captured features are passed to a fully connected layer with SoftMax output to compute the probability distribution over the five big factors (i.e. E, A, C, N, O).

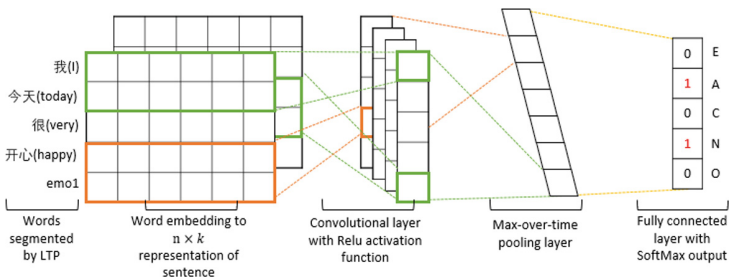


Fig. 2. The network architecture of the text classifier of BFP.

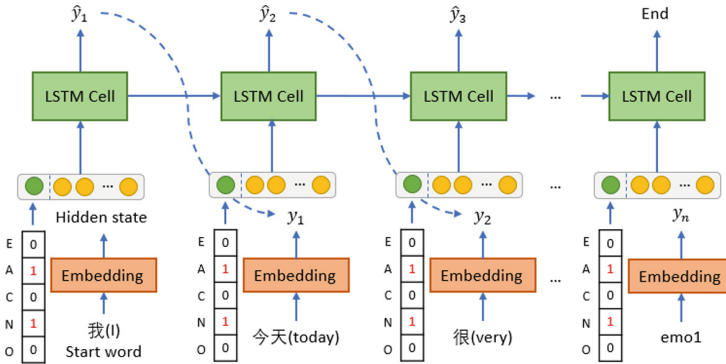
The prepared BFP-labelled Chinese corpora was randomly shuffled and then split into train and validation datasets with ratio of 9:1 for cross-validation of the model parameters. The classification accuracy in each BFP dimension is shown in Table 1. Such a high accuracy demonstrates the reliability of this text classifier, which will be used to label larger social media text corpora for training language generation model.

**Table 1.** The prediction accuracies of validation datasets of the classification model.

Dimension	Accuracy
Extraversion	0.9608
Agreeableness	0.9508
Conscientiousness	0.9713
Neuroticism	0.9806
Openness	0.9640

## 2.2 The Text Generator

The network architecture of text generator, as shown in Fig. 3, comprises an unrolled LSTM sequence. A word-to-vector embedding layer is firstly used to convert input Chinese words into dense tensors. Parameters of embedding layer will be fine-tuned along with training of the generation model.



**Fig. 3.** The network architecture of the text generator.

The output word at time  $y_t$  is predicted by the LSTM cell memory and previous output word  $y_{t-1}$  combined with a global contextual BFP vector of five dimensions, where 1 or 0 is used to represent high or low BFP polarities. The output state of LSTM cell is multiplied by a fully connected layer and eventually a SoftMax layer to determine the occurrence probability distribution of each word in vocabulary. In the training process, the cross-entropy loss between output word sequences and the targets is used for model optimization.

During the process of decoding, a picked first word or sentence should be input to start sequence generation. In practice, a pool of random words was used to select the seed word to enhance the generation variety. The generation loop will continue until the predefined maximum length is reached, or be forced to stop when repeating words or phrases are detected.

### 3 Text Generation Results

Short texts of 6403 Sina Weibo users were selected as training corpora for our conditional text generation model. In order to guarantee the BFP discriminability of training corpora, invalid user IDs including advertisement accounts were cleaned according to predefined filtering criteria. After classified by the above CNN-based text classification model, preprocessed short texts and corresponding BFP representations were then prepared to train the conditional text generator.

#### 3.1 Newly-Generated Text Samples

Text samples from our language generation model under condition of Extraversion dimension are shown in Table 2. The appropriate emoticons have been included to enhance the vividness of generated texts.

**Table 2.** Generated text samples under condition of Extraversion dimension.

Low Extraversion case	High Extraversion case
世界上最难的东西，是一定要自己承担的，也会有结果。	感谢我们的工作人员辛苦了 ❤️❤️❤️
每个人都有自己的痛苦，也许你觉得一个人生活没有什么，但人生如此艰难。	听了好几遍 😊 我的宝贝是不是都好可爱啊! 🤔
每个人都有自己的委屈，所以，我们才能对自己善良。	谁的人生，总是在令人心疼 😭

According to the below table, it can be observed that texts generated under condition of low extraversion are more objective and philosophical and does not exhibit too intense feelings with less emoticons. Oppositely, texts generated under condition of high extraversion are with strong feeling and more emoticons. These results are consistent with our common understanding of human personality. Introverts often like thinking deeply while outgoing people like to pour out their emotions. In addition, generated texts are fluent and logically correlated with proper emotions, which will be demonstrated in Sect. 3.3.



### 3.2 The Human Evaluation of Generated Texts

Since NLG for novel text generation is totally different from machine translation, where BLEU, METEOR and other indicators can be used for objective evaluation, human judgement is employed here to evaluate newly-generated short texts. Similar to the previous work [9], 30 real micro-blog texts and 30 model-generated ones from conditional and unconditional models are mixed and scored by 50 volunteers. Once regarded as human composition, it gets +1 human-like score, otherwise 0. At the meantime, volunteers are asked to mark coherence scores (1–5) of all texts, according to their fluency and logicity. The average scores of model-generated texts and real ones are calculated and compared in Table 3.

**Table 3.** The average scores of model-generated and real texts.

Method	Human-like score	Coherence score
BFP-conditioned generation	<b>0.5387</b>	<b>3.972</b>
Unconditional generation	0.4827	3.536
The real texts	0.7360	4.201

As shown in the above table, human-like and coherence scores of texts from BFP conditional generation model is comparable to real ones, which indicates that our generation system performs comparably to real human compositions in terms of fluency and logicity. As a consequence, our proposed BFP-based psychologically conditional generation model can applied to other NLG missions.

### 3.3 The BFP Scores of Generated Texts

With different BFP input conditions, 50000 short texts are totally generated. The BFP scores of model-generated texts are computed by SC-LIWC psychological analysis method which has been demonstrated effective. To verify the personalized effects of our proposed conditional generation model, we calculated the percentage of newly-generated texts with BFP scores located in three levels (i.e. low, medium, and high), which were determined by using the same standard mentioned in Sect. 2.1. Moreover, the corresponding percentage of machine-composed texts from unconditional generation model is also shown as a reference in Table 4.

As observed in Table 4, the distribution of BFP scores of newly-generated texts from conditional generation model are consistent with their corresponding BFP input conditions, while the BFP scores of newly-composed texts from unconditional generation model almost distribute normally. Here, we define generation accuracy as consistency between computed BFP score level and input BFP condition of newly-generated texts. As a result, the average generation accuracy in each dimension of our proposed conditional generation model is 71.62%, which could be difficultly achieved by traditional unconditionally NLG methods.

**Table 4.** The percentage of newly-generated texts with BFP scores located in three levels.

Dimension	Condition	The newly-generated texts with BFP scores located in		
		Low level	Medium level	High level
Extraversion	Low condition	90.91%	3.50%	5.59%
	High condition	4.30%	19.35%	80.65%
	Unconditional	35.67%	32.48%	31.85%
Agreeableness	Low condition	58.76%	41.24%	0.00%
	High condition	11.24%	37.08%	51.68%
	Unconditional	18.47%	72.61%	8.92%
Conscientiousness	Low condition	67.16%	26.87%	5.97%
	High condition	4.79%	31.51%	63.70%
	Unconditional	24.21%	61.78%	14.01%
Neuroticism	Low condition	66.02%	33.98%	0.04%
	High condition	1.54%	17.69%	80.77%
	Unconditional	10.83%	45.22%	43.95%
Openness	Low condition	75.56%	23.33%	1.11%
	High condition	1.00%	18.00%	81.00%
	Unconditional	25.48%	26.75%	47.77%

## 4 Conclusions and Limitations

In this paper, we have proposed and demonstrated a personalized short text generation model, which consists of a CNN-based text classifier and a conditional LSTM-based text generator. Among them, the CNN-based text classifier has been trained in supervise mode to achieve BFP polarity of social media short texts. With BFP feature vectors as contextual input, a conditional LSTM-based text generator has been trained to compose short texts with discriminative personality styles. Additionally, human evaluation has been conducted to prove the fluency and logicity of the generated texts. Moreover, BFP traits of newly-generated texts have computed based on the SC-LIWC BFP prediction method. High BFP generation accuracy of our conditional generation model facilitates psychologically controllable NLG in some promising industrial applications of artificial intelligent. For example, human-like chat-bots can be benefited from our technique, which can generate personality-specific dialogues according to users' background profiles built with the BFP prediction model.

However, there still exist several limitations in our study. For instance, our generation model based on LSTM is suitable for short text generation such as micro-blog texts, but hardly composes semantically-correlated long sentences and paragraphs. Additionally, in case of instant dialogue systems, mood states of users should be taken account of for automatic dialogue generation in terms of empathy effects.

**Acknowledgement.** This work is supported by Chinese National Natural Science Foundation (61471049, 61532018).

## References

1. Wen, T.H., Gasic, M., Mrksic, N., Su, P.H., Vandyke, D., Young, S.: Semantically conditioned LSTM-based natural language generation for spoken dialogue systems (2015). arXiv preprint: [arXiv:1508.01745](https://arxiv.org/abs/1508.01745)
2. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate (2014). arXiv preprint: [arXiv:1409.0473](https://arxiv.org/abs/1409.0473)
3. Rush, A.M., Chopra, S., Weston, J.: A neural attention model for abstractive sentence summarization (2015). arXiv preprint: [arXiv:1509.00685](https://arxiv.org/abs/1509.00685)
4. Vinyals, O., Toshev, A., Bengio, S., Erhan, D.: Show and tell: a neural image caption generator. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3156–3164. IEEE (2015)
5. Karpathy, A.: The unreasonable effectiveness of recurrent neural networks. Andrej Karpathy blog (2015)
6. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: Advances in Neural Information Processing Systems, pp. 3104–3112 (2014)
7. Guo H.: Generating text with deep reinforcement learning (2015). arXiv preprint: [arXiv:1510.09202](https://arxiv.org/abs/1510.09202)
8. Zhang, Y., Gan, Z., Carin, L.: Generating text via adversarial training. In: NIPS Workshop on Adversarial Training, vol. 21 (2016)
9. Li, J., Galley, M., Brockett, C., Spithourakis, G.P., Gao, J., Dolan, B.: A persona-based neural conversation model (2016). arXiv preprint: [arXiv:1603.06155](https://arxiv.org/abs/1603.06155)
10. Quercia, D., Kosinski, M., Stillwell, D., Crowcroft, J.: Our twitter profiles, our selves: predicting personality with twitter. In: 2011 IEEE Third International Conference on Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third International Conference on Social Computing (SocialCom), pp. 180–185. IEEE (2011)
11. Bai, S., Hao, B., Li, A., Yuan, S., Gao, R., Zhu, T.: Predicting big five personality traits of microblog users. In: Proceedings of the 2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT), vol. 1, pp. 501–508. IEEE Computer Society (2013)
12. Qiu, L., Lu, J., Ramsay, J., Yang, S., Qu, W., Zhu, T.: Personality expression in Chinese language use. *Int. J. Psychol.* **52**(6), 463–472 (2017)
13. Gao, R., Hao, B., Li, H., Gao, Y., Zhu, T.: Developing simplified chinese psychological linguistic analysis dictionary for microblog. In: Imamura, K., Usui, S., Shirao, T., Kasamatsu, T., Schwabe, L., Zhong, N. (eds.) BHI 2013. LNCS (LNAI), vol. 8211, pp. 359–368. Springer, Cham (2013). [https://doi.org/10.1007/978-3-319-02753-1\\_36](https://doi.org/10.1007/978-3-319-02753-1_36)
14. Goldberg, Y., Levy, O.: Word2vec explained: deriving mikolov et al.’s negative-sampling word-embedding method. Eprint Arxiv (2014)



# Deep Multi-modal Learning with Cascade Consensus

Yang Yang, Yi-Feng Wu, De-Chuan Zhan<sup>(✉)</sup>, and Yuan Jiang

National Key Laboratory for Novel Software Technology,  
Nanjing University, Nanjing 210023, China  
{yangy, wuyf, zhandc, jiangy}@lamda.nju.edu.cn

**Abstract.** Multi-modal deep learning has achieved great success in many applications. Previous works are mostly based on auto-encoder networks or paired networks, however, these methods generally consider the consensus principle on the output layers and always need deep structures. In this paper, we propose a novel Cascade Deep Multi-Modal network structure (CDMM), which generates deep multi-modal networks with a cascade structure by maximizing the correlations between each hidden homogeneous layers. In CDMM, we simultaneously train two nonlinear mappings layer by layer, and the consistency between different modal output features is considered in each homogeneous layer, besides, the representation learning ability can be forward enhanced by considering the raw feature representation simultaneously for each layer. Finally, experiments on 5 real-world datasets validate the effectiveness of our method.

**Keywords:** Multi-modal learning · Deep learning · Cascade structure

## 1 Introduction

In most real-world data analysis problems as image processing, medical detection and social computing, complicated objects can always be described from diverse domains and are naturally with multi-modal feature presentations. However, the representations of various modalities are quite different from each other and it is a challenge to fuse the multiple modalities directly with large discrepancy. Recently, substantial efforts have been dedicated to consider the modal consensus problem, which generally maximizes the correlation between different modalities in the projected subspace. Modern multi-modal subspace learning methods mainly derived from the CCA method [7]. However, these methods are most linear ones, though they can be extended to non-linear models with kernel tricks as KCCA [1], it is difficult to design a suitable kernel and also inefficient to deal with the large datasets.

---

This work was supported by the National Key R&D Program of China (2018YFB1004300), NSFC (61773198, 61632004).

Recently, multi-modal methods based on deep networks have attracted more attention, which more easily to process large amounts of data [9, 12]. Different from KCCA, these methods generally maximize the correlation between the output features of multiple distinct modal networks for learning more discriminative feature representations. Though deep networks are powerful, it is notable that the structures of deep CCA are very complicated and always require deeper structure for better representation learning, while leaving the consensus principle of the homogeneous hidden layers among different modal networks without considering during the training phase. Thus, in recent, [19] proposed the gcForest, which generates a deep forest ensemble method with a cascade structure, it is notable that the number of cascade levels can be adaptively determined such that the model complexity can be automatically set.

Inspired by this fact, we therefore propose the CDMM (Cascade Deep Multi-Modal networks) approach to learn multiple maximal correlated deep networks simultaneously, which trains the multiple deep networks with a cascade structure by maximizing the correlation between each homogeneous layers of different modal networks. Specifically, we train multiple deep nonlinear networks layer by layer, and consider the consistency between each homogeneous layer of different modal networks carefully, and then output the processing result to the next level without retraining anymore. As a consequence, the number of layers can be adaptively determined. On the other hand, we forward enhance the network representational learning ability by concatenating the raw input with the output of each hidden layer.

## 2 Related Work

The exploitation of multiple modal subspace learning has attracted many attentions recently. Most proposed methods are mainly derived from the CCA methods, which are devoted to fully utilize the relationships between multiple modalities, and leveraging the consistency among different modalities is one of the significant principles. CCA style subspace learning approaches have been well developed in decades [3, 14, 15]. However, these methods are most linear ones. Thus, Kernel canonical correlation analysis (KCCA) [1] extended the CCA to nonlinear projections. Nevertheless, these methods are limited by the fixed kernel and are difficult to handle a large amount of data.

Therefore, considering deep networks can learn nonlinear feature representations without suffering from the drawbacks of nonparametric models, and have achieved great success in many scenarios [10, 16, 20]. Recently [2] used the DCCA to learn complex nonlinear transformation for two modalities; [17] proposed the DCCAЕ, which combined the DCCA and deep auto-encoder in one unified framework for more discriminative feature representation. All these methods employ the deep neural network to maximize the correlation on the output feature representation of multiple distinct modalities. Nevertheless, they only expect the output feature representations of different distinct modal networks

to be maximally correlated, which need deeper networks for learning better discriminative features, while ignoring the correlations among the homogeneous hidden layers.

To the best of our knowledge, previous linear or kernelized multi-modal methods, which improved the performance by considering the consistency among different modalities on the projected subspace, are difficult to handle a large amount of data and are restricted to the reproducing kernel Hilbert space. Though deep CCA based methods solved these problem, yet they only consider the consensus principle of the output feature representation. In this paper, we propose the CDMM (Cascade Deep Multi-Modal networks), which trains multiple separate deep network with the cascade structure by considering the consistency between homogeneous hidden layers of different modalities layer by layer, moreover, the representation learning ability can be forward enhanced gradually by considering the raw input for each layer. Consequently, we can obtain a competitive performance with a controlled number of hidden layers.

### 3 Proposed Method

Suppose we have  $N$  instances, denoted by  $\mathcal{D} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_i, \dots, \mathbf{x}_N\}$ , where each instance  $\mathbf{x}_i = [x_{i_1}, x_{i_2}, \dots, x_{i_d}] \in \mathcal{R}^d$ . Meanwhile, in multi-modal learning, instance space can be denoted as  $M$  parts without overlap,  $v = \{v_1, v_2, \dots, v_M\}$ , where  $\mathbf{x}^{v_i} \in \mathbb{R}^{d_i}$  is raw features from the  $i$ -th modality,  $d = d_1 + d_2 + \dots + d_M$ . Without any loss of generalities, each instance  $\mathbf{x}_i$  can be denoted as  $(\mathbf{x}_i^{v_1}, \mathbf{x}_i^{v_2}, \dots, \mathbf{x}_i^{v_M})$ .

#### 3.1 Deep Canonical Correlation Analysis (DCCA)

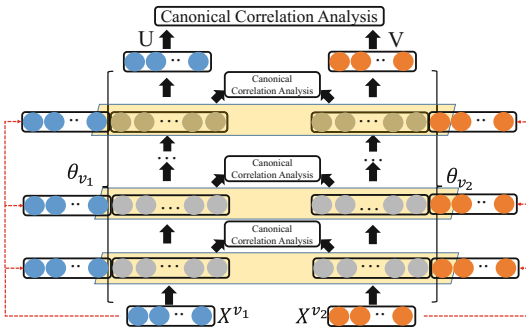
Recently, several works are proposed to combine the deep neural network and CCA for better feature representation learning, [2] proposed the deep canonical correlation analysis (DCCA) approach. In DCCA, two deep neural networks  $f_{v_1}$  and  $f_{v_2}$  are used to extract nonlinear features for different modalities, and then maximize the canonical correlation between the extracted features  $f_{v_1}(X^{v_1})$  and  $f_{v_2}(X^{v_2})$ , which can be represented as:

$$\begin{aligned} \max_{\theta_{v_1}, \theta_{v_2}, U, V} \quad & \frac{1}{N} \text{tr}(U^\top f_{v_1}(X^{v_1}) f_{v_2}(X^{v_2})^\top V) \\ \text{s.t.} \quad & U^\top \left( \frac{1}{N} f_{v_1}(X^{v_1}) f_{v_1}(X^{v_1})^\top + r_{v_1} I \right) U = I, \\ & V^\top \left( \frac{1}{N} f_{v_2}(X^{v_2}) f_{v_2}(X^{v_2})^\top + r_{v_2} I \right) V = I, \end{aligned} \quad (1)$$

where  $\theta_{v_1}$  and  $\theta_{v_2}$  are the weight parameters of networks  $f_{v_1}$  and  $f_{v_2}$ ,  $U$  and  $V$  are the CCA directions which project the output features to the same subspace.  $(r_{v_1}, r_{v_2}) > 0$  are regularization parameters for same covariance estimation [4], the  $U^\top f_{v_1}(X^{v_1})$  and  $f_{v_2}(X^{v_2})^\top V$  are the final projection mapping for testing. Nevertheless, DCCA method and the extensions most concentrate on the correlation between the output feature representation, while ignoring the correlation between homogeneous hidden layers.

### 3.2 Cascade Deep Multi-Modal Networks (CDMM)

In this section, we mainly introduce the concrete steps on learning the discriminative deep multi-modal feature representations with a novel cascade structure, which takes the consensus principle into consideration for each homogeneous hidden layer. We simultaneously train paired deep networks layer by layer, and maximize the consistency between the homogeneous output feature representation of the hidden layers, consequently, we can learn more discriminative feature representations for different modalities, meanwhile, the layers of different modal networks can be adaptively induced by the performance measure, rather than designed in advance manually. On the other hand, the estimated output of hidden layer forms a feature representation vector, which is then concatenated with the raw feature vector to be the input of the next cascade layer for more robust feature representation.



**Fig. 1.** The overall flowchart. CDMM consists two homogeneous deep networks, which trains with cascade structure different from previous DNN-based method. During the training phase, CDMM maximally correlates each homogeneous hidden layer, besides, the raw features are concatenated with each hidden layer output as next input for more robust representations.

connection matrix for each layer as the DNN-based multi-modal representation learning models as [2], which can be further implemented to convolution structure as CNN-based model. Then, in order to maximize the correlation of each homogeneous layer of different modalities, we consider the hidden layer output (shown in yellow shadows) of each modal networks as the  $f_{v_1}(X^{v_1})$  and  $f_{v_2}(X^{v_2})$  in Eq. 1, and optimize the parameters of current hidden layers as the DCCA.

It is notable that the objective couples all training samples through the whitening constraints, so stochastic gradient descent (SGD) cannot be applied in a standard way, yet it has been observed by [2] that DCCA can still be optimized efficiently as long as the gradient is estimated using a sufficiently large minibatch. Intuitively, this approach works due to a large minibatch contains

Representation learning in deep neural networks mostly relies on the layer by layer processing of the raw features. Inspired by this recognition, [19] proposed the gcForest, which employs a cascade structure, where each layer of the cascade structure receives feature information processed by its preceding level, and output its processing result to the next level. Thus, we propose a novel deep multi-modal networks with the cascade structure as shown in Fig. 1. Specifically, CDMM can be with different deep structure, and for simplicity, we use fully connection

enough information for estimating the covariances. Then, the outputs of the estimated hidden layers form a feature representation, considering the representations of the shallow layers of the deep network structure are usually weak features, the hidden layer output is then concatenated with the raw feature vector to be input to the next level of cascade as shown in Fig. 1, i.e., the dimension of the hidden layer output is 1024, and the raw feature is 798 dimensionality, thus, the next level of cascade will receive 1822 ( $= 1024 + 798$ ) augmented features. It is notable that the transformed feature vectors, augmented with the raw feature representations, will then be used to train the next grade of cascade multi-modal networks respectively, and the parameters of preceding hidden layers remain unchanged.

## 4 Experiments

### 4.1 Datasets and Configurations

CDMM can learn more discriminative multi-modal feature representation with self-adaption networks. In this section, we will provide the empirical investigations and performance comparison of CDMM. In particular, we demonstrate these phenomenon on 5 real datasets, i.e., MNIST generates two modal data using the original MNIST dataset [11]. As in [17], we randomly rotate the images and the resulting images are used as modal  $v_1$  inputs. For each  $v_1$ , we randomly select an image of the same identity from the original dataset, add independent random noise to obtain the corresponding modal  $v_2$  sample; AVLETTER contains 10 speakers speaking the letters A to Z at 3 times for each one. This dataset provides pre-extracted lip regions of  $60 \times 80$  pixels as modal  $v_1$  and audio features (raw audio is not provided) Mel-Frequency Cepstrum Coefficient (MFCC) as modal  $v_2$ ; XRBM follows the setup of [17]. Inputs to multi-modal feature learning are acoustic features as modal  $v_1$ , and articulatory features concatenated over a 7-frame window around each frame as modal  $v_2$ ; WIKI [13] is a rich-text web document dataset with images, which has 2,866 documents extracted from Wikipedia as modal  $v_1$ . Each document is accompanied by an image as modal  $v_2$ . Text is represented by TF-IDF feature with 7343-dimensional; FLICKR8K [6] consists of 8,000 images that are each paired with five different captions, similarly, we denote the image as model  $v_1$  and text information as modal  $v_2$ .

For WIKI and FLICKR8K datasets, 70% instances are chosen as training set, 20% are chosen as validation set and the remains are test set as [18]. In other three datasets, training and test splits are provided by [2, 8]. For DNN-based models, feature mappings  $(f_{v_1}, f_{v_2})$  are implemented by networks of 2 or 3 hidden layers, each of 1,024 sigmoid units, and a linear output layer of L units, we refer to a DNN-based model with an output size of o and d layers (including the output) as \*-o-d, i.e., CDMM-o-d, DCCA-o-d, DCCAE-o-d. The two networks  $(f_{v_1}, f_{v_2})$  are pre-trained in a layerwise manner using restricted Boltzmann machines [5], and SGD is used for optimization with minibatch size as 800, learning rate and momentum tuned on the tuning set, a small weight decay parameter of  $10^{-4}$  is used for all layers.



## 4.2 Comparing with CCA-Based Multi-modal Methods

CDMM is firstly compared to linear and kernelized multi-modal CCA-based methods. Since there are deep networks in CDMM, DNN-based multi-modal methods are also compared in the experiments. In detail, the compared methods are listed as: Linear CCA (CCA), Kernel CCA, DCCA, DCCA-E.

Table 1 compares the total correlation on the test sets obtained for the 10 most correlated dimensions with compared methods. It clearly reveals that on all datasets, with the same number of layers, the CDMM total correlation is the highest. Besides, note that CDMM also has exceeded other compared methods only with 2 layers on most datasets except XRBM. Thus, CDMM can acquire more discriminative feature representation with shallow deep network structures.

**Table 1.** The correlation of CDMM with compared methods. The significant best classification performance on each dataset is bolded.

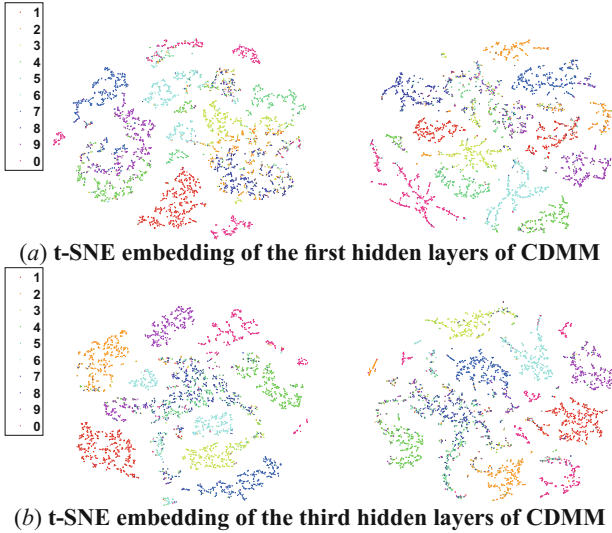
	MNIST	AVLETTER	XRBM	WIKI	FLICKR8K
CCA	3.59	6.55	15.97	5.23	4.87
KCCA	1.29	2.37	35.51	7.25	6.80
DCCA-10-2	7.49	7.21	42.14	10.21	7.04
DCCA-10-3	7.84	7.24	43.00	10.86	7.17
DCCA-E-10-2	7.81	7.37	42.23	10.09	7.08
DCCA-E-10-3	7.94	7.41	42.50	10.80	7.24
CDMM-10-2	8.03	14.11	42.14	11.18	8.04
CDMM-10-3	<b>8.07</b>	<b>14.21</b>	<b>43.20</b>	<b>11.40</b>	<b>8.06</b>

## 4.3 Investigation on Embedding of Different Layers

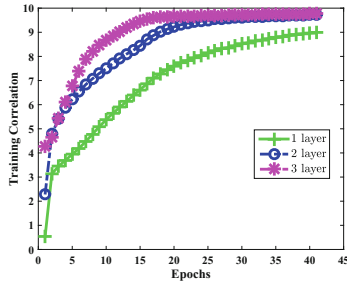
In order to explore the influence of the cascade structure, more experiments are conducted. We qualitatively investigate the features by embedding the projected features in 2D using t-SNE of each pair homogeneous hidden layers, the resulting visualizations are given in Fig. 2. Each sample is denoted by a marker located at its coordinates of embedding and color coded by its label. Due to the page limits, we only list the noisy MNIST digits dataset for verification. From the Fig. 2, we can find that CDMM gives more accurate embedding with the cascade structure from the initial layers, i.e., CDMM pushed different digits far apart from the initial layers.

## 4.4 Empirical Investigation on Convergence

To investigate the convergence of CDMM iterations empirically. The objective function value, i.e., the value of Eq. 1 of CDMM in each iteration of each homogeneous layers are recorded. Due to the page limits, only results on noisy MNIST



**Fig. 2.** t-SNE embedding of the projected MNIST and noisy MNIST digits. Left represents the projected original MNIST modality, and right denotes the noisy MNIST modality. Each sample is denoted by a maker located at its coordinates of embedding and color coded by its label. (Color figure online)



**Fig. 3.** The correlation of different layers in training phase of Noisy MNIST digit dataset.

digits dataset are plotted in Fig. 3. It clearly reveals that the correlation value between different modalities increases as the iterations increase, and the performance is stable after several layers in Fig. 3, i.e., the variations between the correlation values of second hidden layers and third hidden layers less than the predefined threshold, which can be used to control the layers self-adaptively.

## 5 Conclusion

Previous DNN-based multi-modal networks have been used for learning more discriminative feature representations. However, these methods only consider the

consensus principle on output layers and always need predefined the network structures, i.e., number of layers, which lead complex deep network structures and high computation expense, while neglect considering the correlation between the homogeneous hidden layers of different deep modal structures. In this paper, we propose a novel Cascade Deep Multi-Modal networks (CDMM). This method generates a deep multi-modal networks with a cascade structure which fully maximizes the correlations between homogeneous hidden layers of different modal networks, and can acquire representative networks with shallow layers. Besides, the representational learning ability can be further enhanced by concatenating the raw features with each hidden layer output. And empirical studies show that we can learn more discriminative features with shallow layers. How to extend the scalability with improved performance is an interesting future work.

## References

1. Akaho, S.: A kernel method for canonical correlation analysis, pp. 263–269 (2007)
2. Andrew, G., Arora, R., Bilmes, J.A., Livescu, K.: Deep canonical correlation analysis. In: Proceedings of the 30th International Conference on Machine Learning, Atlanta, GA, pp. 1247–1255 (2013)
3. Arora, R., Mianjy, P., Marinov, T.V.: Stochastic optimization for multiview representation learning using partial least squares. In: Proceedings of the 33rd International Conference on Machine Learning, New York, NY, pp. 4847–4855 (2016)
4. Hardoon, D.R., Szedmak, S.R., Shawe-Taylor, J.R.: Canonical Correlation Analysis: An Overview with Application to Learning Methods. MIT Press, Cambridge (2004)
5. Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. *Science* **313**(5786), 504–507 (2006)
6. Hodosh, M., Young, P., Hockenmaier, J.: Framing image description as a ranking task: data, models and evaluation metrics. *JAIR* **47**, 853–899 (2013)
7. Hotelling, H.: Relations between two sets of variates. *Biometrika* **28**(3–4), 321–377 (1936)
8. Hu, D., Li, X., Lu, X.: Temporal multimodal learning in audiovisual speech recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, pp. 3574–3582 (2016)
9. Kan, M., Shan, S., Chen, X.: Multi-view deep network for cross-view classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, pp. 4847–4855 (2016)
10. Kang, G., Li, J., Tao, D.: Shakeout: a new regularized deep neural network training scheme. In: Proceedings of the 30th AAAI Conference on Artificial Intelligence, Phoenix, Arizona, pp. 1751–1757 (2016)
11. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. *Proc. IEEE* **86**(11), 2278–2324 (1998)
12. Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., Ng, A.Y.: Multimodal deep learning. In: Proceedings of the 28th International Conference on Machine Learning, Bellevue, Washington, pp. 689–696 (2011)
13. Rasiwasia, N., Pereira, J.C., Coviello, E., Doyle, G., Lanckriet, G.R., Levy, R., Vasconcelos, N.: A new approach to cross-modal multimedia retrieval. In: Proceedings of the 18th ACM International Conference on Multimedia, Firenze, Italy, pp. 251–260 (2010)

14. Rupnik, J., Shawe-Taylor, J.: Multi-view canonical correlation analysis. In: Slovenian KDD Conference on Data Mining and Data Warehouses, Ljubljana, Yugoslavia, pp. 1–4 (2010)
15. Shrivastava, A., Rastegari, M., Shekhar, S., Chellappa, R., Davis, L.S.: Class consistent multi-modal fusion with binary features. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, pp. 2282–2291 (2015)
16. Tian, F., Gao, B., Cui, Q., Chen, E., Liu, T.Y.: Learning deep representations for graph clustering. In: Proceedings of the 28th AAAI Conference on Artificial Intelligence, Quebec, Canada, pp. 1293–1299 (2014)
17. Wang, W., Arora, R., Livescu, K., Bilmes, J.: On deep multi-view representation learning. In: Proceedings of the 32nd International Conference on Machine Learning, Lille, France, pp. 1083–1092 (2015)
18. Yang, Y., Zhan, D.C., Jiang, Y.: Deep learning for fixed model reuse. In: Proceedings of the 31st AAAI Conference on Artificial Intelligence, New York, NY, pp. 1033–1039 (2017)
19. Zhou, Z.H., Feng, J.: Deep forest: towards an alternative to deep neural networks. In: Proceedings of the 26th International Joint Conference on Artificial Intelligence, Melbourne, Australia (2017)
20. Zhu, X., Huang, Z., Wu, X.: Multi-view visual classification via a mixed-norm regularizer. In: Pei, J., Tseng, V.S., Cao, L., Motoda, H., Xu, G. (eds.) PAKDD 2013, Part I. LNCS (LNAI), vol. 7818, pp. 520–531. Springer, Heidelberg (2013). [https://doi.org/10.1007/978-3-642-37453-1\\_43](https://doi.org/10.1007/978-3-642-37453-1_43)



# Driving the Narrative Flow of an Interactive Storytelling System for Case Studies

Stanley Yu Galan, Michael Joshua Ramos, Aakov Dy, Yusin Kim, and Ethel Ong<sup>(✉)</sup>

De La Salle University, 2401 Taft Avenue, Manila, Philippines  
{stanley\_yugalan, ethel.ong}@dlsu.edu.ph

**Abstract.** Interactive Storytelling systems combine natural language processing techniques with gameplay to provide a narrative-based environment where a player's decisions can alter the flow of the game. This type of simulated story world setup can facilitate the teaching and learning of concepts in different domains, such as the case method. A case method is an approach utilized by educators to encourage student participation through the direct discussion of a given specific real-life or imagined situation (the case study) where they can practice their decision-making skills. In this paper, we discuss how we designed an interactive storytelling environment with case studies as its central theme, to enable business students to learn ethical business practices and simulate the effect of their decisions to the organization.

**Keywords:** Interactive storytelling · Case studies · Story generation  
Domain knowledge base

## 1 Introduction

Interactive storytelling combines story generation techniques with game-based learning principles to encourage readers (or *players*) to participate in the generation of stories by allowing their in-game decisions to affect the progression of the narrative being told [1]. The reader enters the virtual world by portraying as one of the characters who pursue actions toward the achievement of a certain goal [2]. A notable example is *Façade* [3], where the player assumes the role of a close friend of a couple who are experiencing domestic problems. Using natural language processing techniques, the player interacts with the couple and this interaction may affect the outcome of the confrontation of the two story characters.

Interactive storytelling has also been applied in childhood education. *Adventures of Ellie* [4] is designed for children with autism to learn proper social behavior, such as *greeting others*, *waiting for your turn*, *sharing*, and *being tidy*, through social stories [5]. The use of a simulated environment affords these children an opportunity to develop their problem-solving skills with the help of a virtual peer who explains the consequences of user actions and the rationale behind every desirable action.

In the classroom, educators design case studies to encourage students to participate in the direct discussion of the given scenarios. This case method [6] approach gives learners a venue to analyze specific real-life or imagine situations, and practice decision

making skills in an environment with few consequences [7]. Interactive storytelling systems can be explored as a platform for the delivery of these case studies.

This paper presents the design of Pizzeria Story, an interactive storytelling system set in a business world to allow students taking introductory business courses to learn about the effects of ethical business practices. We discuss the representation of storytelling knowledge as a semantic ontology sourced from deconstructing a corpus of cases on businesses and stored as assertions in Prolog. We also describe the interaction between the front-end visual game engine and the back-end story generator to impart the necessary business concepts and practices to the player.

## 2 Storytelling Knowledge

Culliton [8] explains that a case is a story about an incident or how people handle a situation. Naumes and Naumes [7] further define a case as the written story of a company, an institution or a situation. A case has a structure and sufficient details to enable the reader to picture what is being described. It includes elements commonly found in stories, namely the *theme*, *setting*, *character*, *events* and *plot*.

The theme is the message for the reader to take away and continue to think about once the story has ended. Cases involving organizations can contain themes on ethical business issues, such as abusive behavior, employee rights, misconduct, conflict of interest, stealing, and misuse of company resources. These ethical issues are based on the common good principle that calls for the organization of the social economy such that its members realize common interest in the provision of certain basic goods to all members of the community [9].

The setting describes the time and location where the story takes place, such as the company headquarters, employees' workplace and corporate sites where business transactions may occur. The plot is a planned, logical series of events that involve stakeholders. A common plot structure contains the exposition that introduces the setting and characters, the complication that comprises a series of events depicting the conflict (ethical dilemma), and the climax or resolution of the conflict.

### 2.1 Event Model

An event can be of two types: the actions performed by a character, such as *cooking food* and *preparing reports*; or their outcome, such as *food is cooked*, which can lead to another event, e.g., *servicing the food*. A character, which may be the organization and its stakeholders (managers, employees, customers and suppliers), takes part in the actions that occur in a story, either as the doer or the recipient of an action's outcome.

Events may be annotated with a set of pre-conditions that dictate when these can be selected as part of the options presented to the player, and a set of post-conditions that describe changes to the story world when these events have taken place. Table 1 provides a simplified event model that has been adapted from the work of [10]. Variables are indicated with the underscore symbol “\_” preceding the variable name. Data from the story world model is represented as an assertion and appears in italicized font.

**Table 1.** Event model

Event name	Leave
Plot type	employee complaint
Parameters	<i>_agent</i>
Pre-conditions	$role(\_agent, employee) \wedge morale(\_agent, < 20) \wedge$ $capableAction(employee, quit)$
Post-conditions	$employeeCount(pizzeria, -1)$

As shown in Table 1, event *leave* requires a parameter, *\_agent*, as the doer of the action. The plot type categorizes the event as an “employee complaint”. For the event to be selected by the story planner, three (3) conditions must be satisfied - the *\_agent* is an employee, his/her moral is below 20, and he/she is capable of performing the action *quit*. The first two constraints are retrieved from the story world model. The post-condition states that the business will lose an employee (*employee count is decremented*) when this event takes place in the story world.

## 2.2 Domain Knowledge Base

A domain knowledge base (KB) is used to model real-world business concepts, facts and rules that do not change across stories, as shown in Table 2. Domain knowledge is represented as binary assertions of the form  $relation(concept1, concept2)$ , where *relation* denotes the semantic relationship between the two concepts. Assertions, currently numbering 657, are represented as Prolog facts in Pizzeria Story.

**Table 2.** Semantic relations with sample assertions

Relation	Definition	Sample assertions
desires	Denotes the goals or desires of a story character	desires(employee, high salary) desires(manager, high income)
capableAction	States the task to be performed by a given job position	capableAction(waiter, serve) capableAction(employee, report)
receiveAction	Indicates actions that may be performed on a character or object	receiveAction(customer, serve) receiveAction(food, cook)
usedFor	Indicates the purpose of an object	usedFor(tray, serve)
hasProduct	Denotes the products that a particular business may carry	hasProduct(dairy company, milk) hasProduct(dairy company, cheese)
hasProperty	Describes the attributes of a concept	hasProperty(employee, punctual)
isSuperior	Indicates the hierarchy level between two job positions	isSuperior(manager, waiter) isSuperior(owner, manager)
causeEvent	Denotes the causal relationship between two events, <i>event1</i> and <i>event2</i>	causeEvent(harass, report) causeEvent(harass, quit)
isA	Specifies the type of a given concept	isA(waitress, position) isA(milk, dairy product)

Assertions describing more general relations (*isA*, *capableAction*, *receiveAction*, *hasProperty*, *causeEvent*) were adapted from ConceptNet [11] while others expressing specific business concepts (*desires*, *hasProduct*, *isSuperior*) were formulated specifically for the system. The causal relationship between events is organized and represented using the *causeEvent* relation. Characters are also modelled as the actor or doer of the event using the *capableAction* relation, or the recipient of an event’s outcome using the *receiveAction* relation. The objects include the instruments that are needed to perform an action (*usedFor*), or the recipient of the action (*receiveAction*).

### 2.3 Story World Model

The story world model, shown in Fig. 1, is used to track the states of characters and objects as actions and events occur in the story world. These states are considered by the story planner to determine the next action or event that may take place.

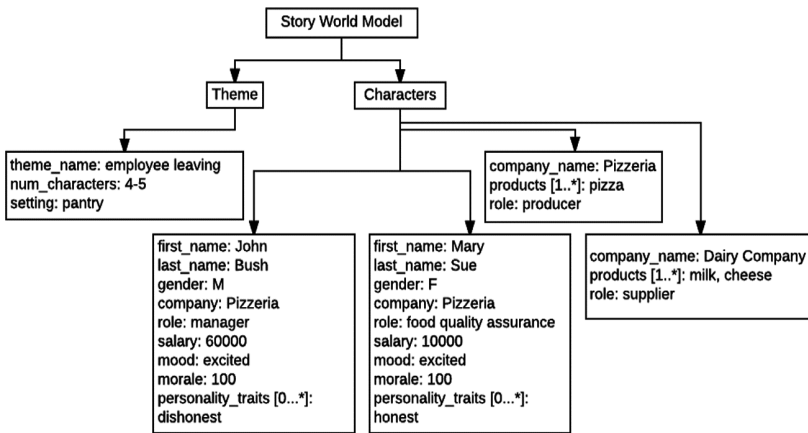


Fig. 1. Story world model

Each character is modelled with a name, gender, company, role, salary, mood, morale level, and list of personality traits. Characters that represent organizations have the attributes *name*, *list of products*, and *role* (i.e., supplier). These attributes are initialized at the start of a story by retrieving assertions from the domain KB. For example, the products of *Dairy Company* are determined from *hasProduct(dairy company, dairy product)*, *isA(milk, dairy product)* and *isA(cheese, dairy product)*. Dynamic attributes, such as *salary*, *mood* and *morale* are updated as the story progresses.

The story world model also contains the story theme, setting and number of characters needed. The theme is used to define character roles and is also used to determine the sequence of events to achieve the target ethical issue that the student will practice his/her decision-making skills on.



### 3 Planning the Case Narrative

Pizzeria Story is a 3D tycoon style game that places the player in charge of running a pizzeria with a goal of keeping the business afloat. To do so, the player must manage employees, product line (pizza) and supplies; and deal with workplace incidents depicted through the interactive cases. The front-end game engine coordinates with the back-end story generator to dynamically generate various business scenarios that the player encounters in the story world. The coordination proceeds in a cycle of story planning – surface text generation – story world update.

Planning the case narrative commences with the random selection of a theme and the instantiation of the story characters. Consider the theme *sexual harassment* shown in Table 3. The planner generates the exposition plot unit that contains a set of assertions to describe the setting; and for each instantiated character, the characters' gender, where they work and roles. A partial plot unit is shown in Listing 1.

**Table 3.** Theme representation

Theme Name	Sexual Harassment
No of Characters (N)	3
Theme Constraints	<p>For each character <math>_C_i</math> (where <math>i &lt; N</math>):</p> $\text{person}(_C_i) \wedge \text{role}(_C_i, \_Role_i) \wedge \text{worksAt}(_C_i, \_Org)$ $\text{isSuperior}(\_Role_3, \_Role_1) \wedge \text{isSuperior}(\_Role_3, \_Role_2) \wedge$ $\text{rank}(\_Role_1) == \text{rank}(\_Role_2) \wedge$ $\text{gender}(_C_1, \text{female}) \wedge \text{gender}(_C_2, \text{male})$

Listing 1. Plot unit for the exposition.

- 
- (1) location('pantry').
  - (2) person('Mary'). person('John'). person('Michael').
  - (3) hasGender(person('Mary'), gender('Female')).
  - (4) worksAt(person('Mary'), organization('Pizzeria')).
  - (5) hasRole(person('Mary'), role('cashier')).
- 

To generate the events that describe the conflict, the planner turns to a story graph [12] representing a branching structure of events that has been pre-scripted at design time. As depicted in Fig. 2, nodes are events and decision points that branch out to multiple nodes. Inbound edges are the pre-conditions while outbound edges contain post-conditions. Every possible path through the graph represents a case narrative.

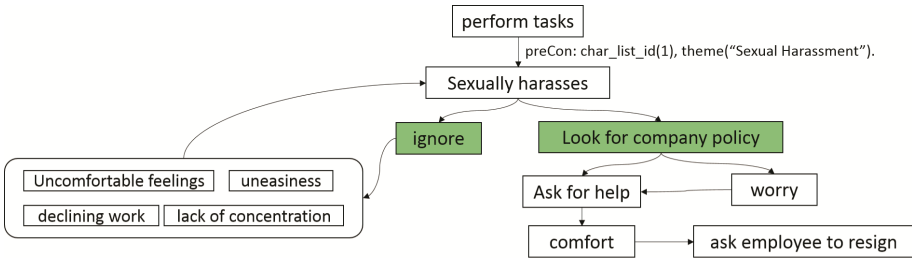


Fig. 2. Sample story graph

The player is given control over how to resolve the ethical dilemma by selecting from a set of candidate actions. The planner determines this set of possible actions by querying the KB using the statement `causeEvent(plot_class(_CAUSE), decision(_EFFECT))`. The `causeEvent` semantic relation is used to find candidate events through their causal relationships as defined in the knowledge base. The KB returns a list of decision points as assertions, examples of which are shown in Listing 2.

Listing 2. Assertions depicting decision points.

- 
- (1) `decision('ignore')`.
  - (2) `decision('look for a company policy')`.
- 

Once a player makes a decision at the game interface, the plot type of the decision is sent back to the story planner. The story planner uses this to execute another query to the KB using the `causeEvent` relations. The story graph is traversed to retrieve a plot unit that is considered to be an outcome or effect of the selected decision. The series of events should eventually lead to the climax plot unit which contains assertions to narrate the pivotal moment of the story. Using our sample case, Listing 3 shows the assertions wherein ‘Mary’ decided to resign from the organization.

Listing 3. Plot unit for the climax.

- 
- (1) `capableOf(person('Mary'), action('resign'))`.
  - (2) `receiveAction(person('Michael'), action('resign'))`.
- 

## 4 Results and Analysis

25 undergraduate students taking Applied Corporate Management individually played with Pizzeria Story to validate its potential as an environment for learning about ethical business practices. Each player used the system for 15 min and did an average of four (4) full in-game cycles. The students then gave their individual assessment on the *content* and *language* of the generated stories, and the *game interface* and *gameplay* using a

rating scale of 1 (strongly disagree) to 5 (strongly agree). Table 4 summarizes the average scores of Pizzeria Story from the two iterations of testing.

**Table 4.** Evaluation score of Pizzeria Story

Criteria	Iteration #1	Iteration #2
Story content	3.50	3.62
Language	4.30	3.82
Game interface	3.36	4.11
Gameplay	3.51	3.84

The system received an average score of 3.50 in the *story content* criterion. While the participants found the generated case narratives to be relevant with appropriate settings, the stories themselves lack depth, difficulty and variety of scenarios. Consider the story excerpt in Listing 4. Characters were assigned roles based on the constraints of the selected theme; however, character relations are not properly explained, such as the missing background information on the character John who appeared in line (2). Statements describing John’s actions that can be considered as harassment are also missing. While the evaluators found the flow of events to be logical, the events are somewhat simplistic and occur summarily. In line (7), John was fired without proper investigation nor lead time. The generated stories are also missing events, such as those that transpired after John has been fired. Participants also expressed a desire to be given more decision points and morally ambiguous choices to allow them to have more control over the progression of events and to make difficult ethical decisions, respectively, such as how to deal with the actions of John.

Listing 4. Sample story excerpt on the theme *Sexual Harassment*.

- 
- (1) Mary’s tasks are to clean the restaurant and serve customers.
  - (2) John sexually harasses Mary.
  - (3) Mary was distressed by the incident. What should Mary do?
  - (4) [Player chooses “Look for company policy.”]
  - (5) Mary decides to look for a company policy regarding sexual harassment.
  - (6) Mary decides to ask for help from Sarah regarding what to do.
  - (7) Sarah approach John and asked him to resign.
- 

In the *language* criterion, the participants found the stories easy to understand, with correct grammars and correct usage of words. Story text also sound natural. Most of the errors are due to incorrect verb tenses, missing punctuations and incorrect grammar for the generated dialogues.

For *gameplay*, the players generally felt that the concept had potential but could be improved with more scenarios to motivate analytical thinking. The number of variables that players can control should also be increased to cover not only the setting of employee-related parameters, such as salary, but also parameters relevant to the production process of making pizzas. It is worth noting that participants place equal emphasis on good story content and gameplay, with the suggestions that “*it is more important to*

*improve the story itself and its continuity*” from one story to the next, and *“the main point of improvement is the case generation and its effect on the business itself and the employees”*, on top of visual changes to the game interface.

## 5 Conclusion

Interactive storytelling set in a game-based environment afforded students an opportunity to simulate the outcomes of their decisions in a given case scenario and witness their impact to the organization and its stakeholders. On its own, story generation was insufficient to immerse the players into the story world, but when paired with a 3D game environment, the gameplay provided the needed context to help the players perform decision-making tasks. Consistency between the narrative text and the visual game interface also has a major impact on the players. Assertions provided a uniform mechanism to support the exchange of data between the front-end game engine and the back-end story generator. However, the story world model should be expanded to maintain a historical chain of previously generated events to reduce redundancy.

Feedback from participants showed the potential of Pizzeria Story as a compelling teaching tool in encouraging students to look at case studies more seriously since their decisions have a tangible outcome on the game. It was observed that the generated narratives have a bias towards utilitarian ethics. There is an evident tendency for the system to encourage students to perform actions that provide the most good for the majority even if it involves skirting some ethical rules in the process. Future works can look into extending the contents of the knowledge base to present scenarios that promote other ethical systems, to increase the variety of the generated case stories, and to provide textual feedback as part of the story to explain the effects of the decisions on the game world. The planner should also be designed to generate story situations that are more morally ambiguous to allow players to better use their judgement.

## References

1. Ramirez, A., Bulitko, V.: Automated planning and player modeling for interactive storytelling. *IEEE Trans. Comput. Intell. AI Games* 7(4), 375–386 (2015)
2. Cavazza, M.O., Charles, F., Mead, S.J.: Character-based interactive storytelling. *IEEE Intell. Syst.* 17(4), 17–24 (2002)
3. Mateas, M., Stern, A.: *Façade: An experiment in building a fully-realized interactive drama*. In: *Proceedings of the Game Developers Conference, San Jose, CA* (2003)
4. Ong, E.C.J., Consignado, D.G., Ong, S.J., Soriano, Z.C.: Building a semantic ontology for virtual peers in narrative-based environments. In: Numao, M., Theeramunkong, T., Supnithi, T., Ketcham, M., Hnoohom, N., Pramkeaw, P. (eds.) *PRICAI 2016. LNCS (LNAI)*, vol. 10004, pp. 65–76. Springer, Cham (2017). [https://doi.org/10.1007/978-3-319-60675-0\\_6](https://doi.org/10.1007/978-3-319-60675-0_6)
5. Gray, C.: *The New Social Story Book: 10th Anniversary Edition*. Future Horizons, Arlington (2010)
6. Leenders, M.R., Erskine, J.A.: *Case Research: The Case Writing Process*, 2nd edn. Research and Publications Division, School of Business Administration, University of Western Ontario, London (1978)

7. Naumes, W., Naumes, J.R.: *The Art & Craft of Case Writing*, 3rd edn. M.E. Sharpe, Inc., Armonk (2012)
8. Culliton, J.W.: *Handbook on case writing*. Asian Institute of Management, Makati, Rizal, Philippines (1973)
9. Lutz, M.: *Economics for the Common Good: Two Centuries for Social Economic Thought in the Humanistic Tradition*. Routledge, London (1999)
10. Ang, K., Ong, E.: Planning children's stories using agent models. In: Richards, D., Kang, B.H. (eds.) PKAW 2012. LNCS (LNAI), vol. 7457, pp. 195–208. Springer, Heidelberg (2012). [https://doi.org/10.1007/978-3-642-32541-0\\_17](https://doi.org/10.1007/978-3-642-32541-0_17)
11. Liu, H., Singh, P.: ConceptNet - a practical commonsense reasoning tool-kit. *BT Technol. J.* **22**(4), 211–226 (2004)
12. Riedl, M.O., Young, R.M.: From linear story generation to branching story graphs. In: Young, R.M., Laird, J. (eds.) *Proceedings of the First AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, pp. 111–116 (2005)



# Pum-Riang Thai Silk Pattern Classification Using Texture Analysis

Kwankamon Dittakan and Nawanol Theera-Ampornpunt<sup>(✉)</sup>

Prince of Songkla University Phuket Campus, Phuket, Thailand  
{kwankamon.d,nawanol.t}@phuket.psu.ac.th

**Abstract.** Pum-Riang is a type of Thai silk with many patterns. Only experts can identify these patterns on sight. In order to help the general public who are interested in Pum-Riang silk, we propose an automatic Pum-Riang pattern detection using texture analysis. The process is divided into the feature extraction step, feature extraction step, and classifier training step. For each step, we compare various methods and parameters when applicable. The best model is evaluated on a separate test set. It achieves the perfect accuracy of 1.0, indicating that all test samples are correctly classified.

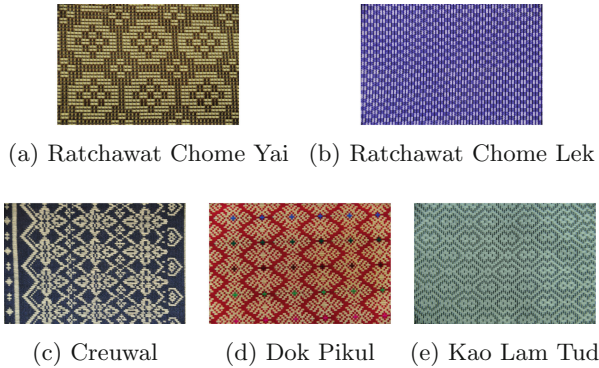
**Keywords:** Image mining · Texture analysis · Image processing

## 1 Introduction

Pum-Riang is a type of Thai silk with elaborate patterns. It is made in Pum-Riang Sub-district in Surat Thani Province in southern Thailand. Historically, clothes made of Pum-Riang Silk is only worn by Thai royalties and aristocrats. Nowadays they can be worn by everyone, although they are typically reserved for special occasions such as weddings or religious ceremonies. Pum-Riang silk is made with specific patterns, some of which are shown in Fig. 1. Because there are numerous patterns available that look alike, only experts can identify the name of the pattern on sight. To help the general public with the identification of Pum-Riang silk pattern, we propose an automated classification using texture analysis.

Many image processing techniques have been developed for various computer vision tasks. For our problem of pattern classification, texture analysis techniques are directly applicable. We use Local Binary Pattern (LBP) as well as two techniques based on LBP as our feature extraction method: Rotated Local Binary Pattern (RLBP) and Complete Local Binary Pattern (CLBP). For feature selection, we compare three methods: Chi-squared, information gain, and gain ratio, as well as find out the optimal number of features. Seven types of classifiers are compared: decision tree, naive Bayes, Bayesian network, averaged one-dependence estimators (AODE), support vector machine, logistic regression, and artificial neural network.

The rest of the paper is organized as follows. Section 2 presents related work. Section 3 presents the design of our Pum-Riang Thai silk pattern classification



**Fig. 1.** Various types of Pum-Riang silk.

framework. Section 4 presents the evaluation methods and results. Finally, Sect. 5 concludes the paper.

## 2 Previous Work

Singh et al. use CLBP, LBP, and Color Coherence Vector (CCV) as the texture analysis method to classify human facial expressions [9]. Multi-class support vector machine is used as the classifier. The overall prediction accuracy is 86.4%, 84.1%, and 75.8% for CLBP, LBP, and CCV, respectively.

Automatic detection of defects on fabrics has been widely studied. Chakraborty et al. propose a method of recognizing and identifying defects in silk fabric [1]. The average intensity of the grayscale image is analyzed, and the image is thresholded to produce a binary image where pixels corresponding to defects have value of 1. Fourier transform is then used to separate the fabric's patterns from the defects. The defects are then classified into one of three categories using artificial neural network: high, medium, and low lousiness. Classification accuracy is very good at 98.56%. Ngan et al. propose wavelet-based methods for defect detection on patterned fabric [8]. The results suggest that a combination of wavelet transform and golden image subtraction method produce the best detection rate. Other techniques used include morphological filters [6] and LBP [10].

Soo Jeon et al. propose a system for automatic recognition of woven fabric patterns using artificial neural network [4]. However, rather than the color pattern, the focus is on the fine weaving patterns such as thread density and warp and weft counts. As patterns in Pum-Riang silk are created using different thread colors, their approach is not directly applicable to our task. Kuo et al. use fuzzy C-means clustering to group fabric weave patterns [5]. However, the resulting clusters have not been evaluated in a classification setting. Furthermore, the photos were obtained using a high-resolution scanner, which makes it less convenient for the user of the system, compared to using a digital camera.

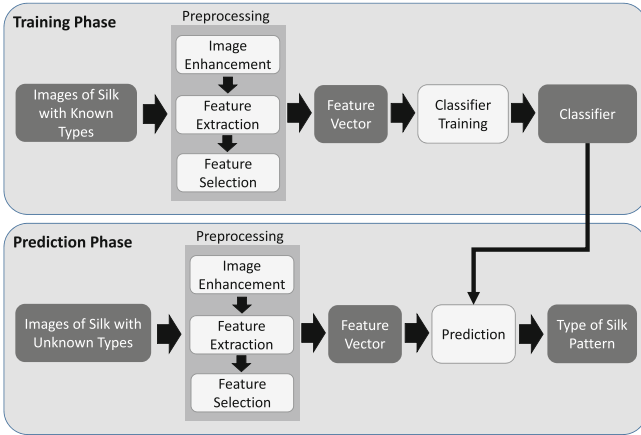


Fig. 2. Overall diagram showing different steps of the image analysis.

### 3 Design

The overview of our framework is shown in Fig. 2. The process can be divided into two overall phases: training phase and prediction phase. In the training phase, the training dataset containing images of silk with known pattern types is used to train a classifier. In the prediction phase, given an image of silk with unknown pattern type, we use the classifier to predict the silk pattern type. In both phases, the raw images first need to go through the preprocessing steps, which consist of image enhancement, feature extraction, and feature selection. The resulting feature vectors are then used to train a classifier using various machine learning methods. We will now describe these steps in more detail.

#### 3.1 Image Enhancement

Patterns used in Pum-Riang silk are made using two main colors. Some patterns such as Dok Pikul contain patterns that utilize additional colors, but these patterns are small when compared to the main patterns. For each pattern, the two main colors may vary. To simplify the later steps, we transform the two main colors to black and white for all images. This is done by thresholding the pixel value. For this work, the threshold is set manually by the user, although it is possible to develop a method that determines the appropriate threshold without human input.

#### 3.2 Feature Extraction

As the number of pixels in an image is in the millions, it is impractical to directly use them as the features to train a classifier. Therefore, we need to extract useful features from the raw pixels. We use and compare the following



three texture analysis techniques which have been successfully applied in other related problems: Local Binary Pattern (LBP), Rotated Local Binary Pattern (RLBP), and Complete Local Binary Pattern (CLBP).

**Local Binary Pattern (LBP).** Local Binary Pattern assigns a value to each pixel, representing the local pattern near that pixel. After assigning a value to every pixel, a vector representing the count of pixels with each unique local pattern is computed and used as the extracted features. It has been found that LBP provides powerful features for texture classification. In this work, we compute LBP using eight adjacent pixels as the neighbors.

**Rotated Local Binary Pattern (RLBP).** Rotated Local Binary Pattern is an extension to LBP to make it rotation invariant [7]. Patterns can be rotated simply because the object is not properly oriented when the photo is taken, and this should not affect the features used for classification. RLBP orients each local pattern by always starting the computation from the neighbor whose difference to the central pixel is maximum. A histogram vector is then computed in the same way as LBP and used as the final feature vector for the image.

**Complete Local Binary Pattern (CLBP).** Both LBP and RLBP capture only the sign of the difference between each pixel and its neighbors. Guo et al. proposed Complete Local Binary Pattern (CLBP) which extends LBP by also including the magnitude of the differences as well as the gray level of the center pixel itself [3].

### 3.3 Feature Selection

After the feature extraction step, we have  $2^N$  features for LBP and RLBP, and  $2 \cdot 2^N + 2$  features for CLBP. Not all features are helpful for classification, as some may capture patterns that are similar across different silk patterns. The goal of the feature selection step is to identify important features and remove the rest in order to reduce model training time and reduce the effects (such as overfitting) that noise in the feature set may have on the prediction accuracy.

There are three main approaches in feature selection: wrapper, filter, and embedded. In the wrapper approach, the model is trained on different subsets of the features and the prediction accuracy is compared. This is computationally expensive and has a risk of overfitting. In the filter approach, a simple filter is used to evaluate and rank the features directly. In the embedded approach, the feature selection method is embedded into the specific model's training algorithm. For this work, we employ the filter approach as it can be used for all classifiers and is not too computationally expensive. Three filters are used and compared: Chi-square, information gain, and gain ratio.

### 3.4 Classifier Training

Once we have the appropriate feature vector for each image, the feature vectors are used to train a classifier. We train and compare 7 different classifiers in order to select the best one. The classifiers compared are decision tree, naive Bayes, Bayesian network, averaged one-dependence estimators (AODE), support vector machine, logistic regression, and artificial neural network. Each classifier's hyperparameters are left at the default settings, according to Weka's implementation.

## 4 Evaluation

### 4.1 Methodology

Our evaluation is separated into three parts: comparing feature extraction methods, comparing feature selection methods, and comparing classifiers. The feature extraction methods are implemented in Matlab while feature selection and classifier training are done using Weka [2].

**Dataset.** We visited fabric stores in Pum-Riang Sub-district in Surat Thani Province in Thailand and took 60 photographs of each of the following five patterns of Pum-Riang silk: Ratchawat Chome Yai, Ratchawat Chome Lek, Creuwal, Dok Pikul, and Kao Lam Tud.

Although the photographs were taken carefully, some are slightly rotated while some others contain reflected light. These imperfections are not explicitly correct as they represent conditions that are likely to happen in the real world, and the methods need to be able to handle them.

For each pattern, 10 samples are separated and used as the test set (50 samples total). The remaining 50 samples for each pattern (250 total) are used as the training set as well as for comparisons of different feature extraction methods, feature selection methods, numbers of features, and classifiers. For these comparisons, 10-fold cross-validation is used as the evaluation method. The metrics reported are area under the ROC curve (AUC), accuracy (AC), sensitivity (SN), specificity (SP), precision (PR), and F-measure (FM).

### 4.2 Feature Extraction

In this section, we compare three feature extraction methods for texture classification: Local Binary Pattern (LBP), Rotated Local Binary Pattern (RLBP), and Complete Local Binary Pattern (CLBP). The feature selection method is fixed as information gain ratio, the number of features after feature selection is fixed as 60, and the classifier used is artificial neural network. The results are shown in Table 1.

Using AUC as the main criteria, CLBP performs best, although the differences are small. This indicates that the magnitude of the difference between neighboring pixels and the center pixel carries important information for prediction.

**Table 1.** Prediction accuracy for each feature extraction method.

Method	AUC	AC	SN	SP	PR	FM
LBP	0.998	0.984	0.984	0.996	0.984	0.984
RLBP	0.993	0.948	0.948	0.987	0.944	0.946
<b>CLBP</b>	<b>1.000</b>	<b>0.980</b>	<b>0.980</b>	<b>0.995</b>	<b>0.980</b>	<b>0.980</b>

**Table 2.** Prediction accuracy for each feature selection method.

Method	AUC	AC	SN	SP	PR	FM
Chi-squared	0.999	0.980	0.980	0.995	0.950	0.980
Information gain	0.999	0.976	0.976	0.994	0.976	0.976
<b>Gain ratio</b>	<b>1.000</b>	<b>0.980</b>	<b>0.980</b>	<b>0.995</b>	<b>0.980</b>	<b>0.980</b>

### 4.3 Feature Selection

In this section, we compare three feature selection methods: chi-squared, information gain, and information gain ratio (abbreviated as gain ratio). The feature extraction method is fixed to CLBP, the number of features is fixed as 60, and the classifier used is artificial neural network. The results are shown in Table 2.

Using AUC as the main criteria, information gain ratio gives the best results, although the differences are small again.

### 4.4 Number of Features

When feature selection is performed, the desired number of features can be controlled by the user. Smaller number of features lead to faster training and prediction, but the prediction accuracy may suffer. In this section, we vary the number of features and compare the prediction performance as well as training time. The numbers of features included in the comparison range from 10 to 100, in increments of 10. The feature extraction method is fixed to CLBP, the feature selection method is information gain ratio, and the classifier used is artificial neural network. The results are shown in Table 3.

Training time grows quickly with the number of features, although there is still some variance. Highest AUC is achieved with 60 features. However, if lower training time is desired, 10 features provide similar prediction accuracy while requiring much lower training time.

### 4.5 Classifier

In this section, we compare the performance of the following classifiers: decision tree, naive Bayes, Bayesian network, AODE, support vector machine, logistic regression, and artificial neural network. The feature extraction method is fixed to CLBP, the feature selection method is information gain ratio, and the number

**Table 3.** Prediction accuracy for different numbers of features, as well as training time taken to build the classifier.

Number of Features	AUC	AC	SN	SP	PR	FM	Training time (seconds)
10	0.999	0.944	0.944	0.986	0.944	0.944	10.36
20	0.998	0.972	0.972	0.983	0.972	0.972	34.68
30	0.999	0.980	0.980	0.995	0.980	0.980	81.56
40	0.999	0.976	0.976	0.994	0.977	0.976	175.78
50	0.999	0.976	0.976	0.994	0.977	0.976	248.56
<b>60</b>	<b>1.000</b>	<b>0.980</b>	<b>0.980</b>	<b>0.995</b>	<b>0.980</b>	<b>0.980</b>	<b>1585.78</b>
70	0.999	0.984	0.984	0.996	0.984	0.984	1921.93
80	0.998	0.976	0.976	0.994	0.976	0.976	3183.28
90	0.998	0.976	0.976	0.994	0.976	0.976	1199.77
100	0.998	0.984	0.984	0.996	0.984	0.984	1927.74

**Table 4.** Prediction accuracy for various classifiers, as well as training time taken.

Classifier	AUC	AC	SN	SP	PR	FM	Training time (seconds)
Decision tree	0.967	0.912	0.912	0.978	0.913	0.912	0.00
Naive Bayes	0.993	0.920	0.920	0.980	0.922	0.919	0.00
Bayesian network	0.994	0.916	0.916	0.979	0.917	0.915	0.00
AODE	0.996	0.952	0.952	0.988	0.952	0.952	0.01
Support vector machine	0.992	0.980	0.980	0.995	0.980	0.980	0.10
Logistic regression	0.978	0.978	0.978	0.960	0.990	0.960	12.70
<b>Artificial neural network</b>	<b>1.000</b>	<b>0.980</b>	<b>0.980</b>	<b>0.995</b>	<b>0.980</b>	<b>0.980</b>	<b>1585.78</b>

of features is 60. All classifiers used are part of the Weka data mining software, with all hyperparameters left at the default values [2]. The results are shown in Table 4.

The classifier that achieves highest prediction accuracy is artificial neural network, with AUC of 1.000. However, it has by far the highest training time. With bigger datasets or more stringent time constraints, AODE and support vector machine may be better choices, as they achieve similar prediction accuracy but with only a fraction of the training time.

#### 4.6 Prediction Performance on Test Set

The previous comparisons are made using 10-fold cross validation on the training set. In order to accurately measure the prediction accuracy, the best model is

evaluated on the held-out test set, which contains 10 samples for each pattern, making the total 50 samples. The best model uses CLBP as the feature extraction method, information gain ratio as the feature selection method, 60 features, and artificial neural network as the classifier.

The model achieves an accuracy of 1.0, indicating that all test samples are classified correctly. This gives us the confidence that the proposed model will be able to classify the five types of Pum-Riang silk accurately. The model can be further developed into an application that anyone can use to find out the type of Pum-Riang silk.

## 5 Conclusion

This paper proposes an automated classification of Pum-Riang Thai silk pattern using texture analysis. The process can be divided into three steps: feature extraction, feature selection, and classifier training. For each step, we compare different methods and parameters in order to find the optimal setting. We find that the best setting overall is CLBP as the feature extraction method, information gain ratio as the feature selection method, number of features equal to 60, and artificial neural network as the classifier. The model achieves the perfect prediction accuracy of 1.0 when evaluated on the held-out test set. This shows that the proposed method is effective for classifying the five types of Pum-Riang Thai silk.

## References

1. Chakraborty, A., Chatterjee, S.M., Kumar, P.K.: Detection of lousiness in silk fabric using digital image processing. In: The 7th International Conference-TEXSCI 2010, pp. 6–8 (2010)
2. Frank, E., Hall, M., Witten, I.: The WEKA workbench. Online Appendix for Data Mining: Practical Machine Learning Tools and Techniques, 4th edn. Morgan Kaufman, Burlington (2016)
3. Guo, Z., Zhang, L., Zhang, D.: A completed modeling of local binary pattern operator for texture classification. *IEEE Trans. Image Process.* **19**(6), 1657–1663 (2010)
4. Jeon, B.S., Bae, J.H., Suh, M.W.: Automatic recognition of woven fabric patterns by an artificial neural network. *Text. Res. J.* **73**(7), 645–650 (2003)
5. Kuo, S.C.Y., Lee, J.Y.: Automatic recognition of fabric weave patterns by a fuzzy C-means clustering method. *Text. Res. J.* **74**(2), 107–111 (2004)
6. Mak, K.L., Peng, P., Yiu, K.: Fabric defect detection using morphological filters. *Image Vis. Comput.* **27**(10), 1585–1592 (2009)
7. Mehta, R., Egiazarian, K.: Rotated local binary pattern (RLBP): rotation invariant texture descriptor. In: 2nd International Conference on Pattern Recognition Applications and Methods, ICPRAM 2013, Barcelona, Spain, 15.-18.2.2013, pp. 497–502. Institute of Electrical and Electronics Engineers IEEE (2013)
8. Ngan, H.Y., Pang, G.K., Yung, S., Ng, M.K.: Wavelet based methods on patterned fabric defect detection. *Pattern Recogn.* **38**(4), 559–576 (2005)

9. Singh, S., Maurya, R., Mittal, A.: Application of complete local binary pattern method for facial expression recognition. In: 2012 4th International Conference on Intelligent Human Computer Interaction (IHCI), pp. 1–4. IEEE (2012)
10. Tajeripour, F., Kabir, E., Sheikhi, A.: Fabric defect detection using modified local binary patterns. *EURASIP J. Adv. Signal Process.* **2008**, 60 (2008)



# Fuzzy Rough Based Feature Selection by Using Random Sampling

Wang Zhenlei<sup>(✉)</sup>, Zhao Suyun, Liu Yangming, Chen Hong,  
Li Cuiping, and Sun Xiran

School of Information, Renmin University of China, Beijing 100872, China  
zhaosuyun@ruc.edu.cn

**Abstract.** Feature selection, i.e., Attribute reduction, is one of the most important applications of fuzzy rough set theory. The application of attribute reduction based on fuzzy rough set is inefficient or even unfeasible on large scale data. Considering the random sampling technique is an effective method to statistically reduce the calculation on large scale data, we introduce it into the fuzzy rough based feature selection algorithm. This paper thus proposes a random reduction algorithm based on random sampling. The main contribution of this paper is the introduction of the idea of random sampling in the selection of attributes based on minimum redundancy and maximum correlation. First, in each iteration the significance of attribute is not computed on all the objects in the whole datasets, but on part of randomly selected objects. By this way, the maximum relevant attribute is chosen on the condition of less calculation. Secondly, in the process of choosing attribute in each iteration, the sample is different so as to select the minimum redundancy attribute. Finally, the experimental results show that the reduction algorithm can obviously reduce the running time of the reduction algorithm on the condition of limited classification accuracy loss.

**Keywords:** Randomly sampling · Fuzzy rough set · Attribute reduction  
Maximum relevance · Minimum redundancy

## 1 Introduction

In recent years, we encounter databases in which the number of objects becomes dramatically large. Hundreds, thousands or even millions of objects are stored in many real-world application databases [1, 2, 5, 10, 11, 24]. Storing and processing all objects might be computationally costly and impractical. To deal with this issue, randomization and randomized algorithms have become feasible and effective tool in machine learning and data mining techniques [1, 11, 14, 22, 23, 28].

Usually, learning algorithm requires a randomized input order for data [2, 23–25]. For instance, in Stochastic Gradient Decent, database records are input into the stochastic gradient decent algorithm in a randomized order. Stochastic Gradient Decent then takes a subset of randomized objects at every iteration to economize the computational cost [12, 27]. Stochastic Gradient Decent, however, has its own limitation. That is, the objective function (i.e., loss function) should be differentiable.

Whereas there are many learning algorithms, such as rough sets, fuzzy rough sets and decision tree, whose objective functions are non-differentiable.

Fuzzy rough set (FRS) is one generalization of rough sets in fuzzy set framework, which assumes that objects characterized by the same information are indiscernible (similar) in the view of the available information about them (with every object in the Universe we associate some information) [26]. The fuzzy indiscernibility (similar/equivalence) relation generated in this way is the mathematical basis of FRS. It makes FRS work well on some problems, but it also limits the further application of FRS. For example, FRS cannot work efficiently on the large-scale datasets because FRS have to discern all the possible pairs in the Universe. As a result, it is promising to propose a new way into FRS which could reduce the size of original problems. Thus, the statistical technique, such as random sampling, may play such a role in rough set theory.

This paper takes the FRS and random sampling as the basic tools to improve the efficiency of FRS on large-scale datasets, and a random sampling based statistical FRS model is then proposed. In this paper we do not directly compute the rough approximation and attribute reduction on the Universe. On the contrary, the random sampling is chosen in the process of computing attribute reduction. Since the selected object size is dramatically smaller than the original size of the analyzed problem, the time and space consumption is significantly reduced.

In the remainder of this paper, Sect. 2 briefly reviews FRS. Section 3 presents some basic concepts which are proposed based on random sampling. Section 4 designs some algorithms based on random sampling to reduce the attributes. And then in Sect. 5 we give some numerical experiments, which clearly demonstrate the effectiveness and efficiency of the proposed method.

## 2 Preliminaries

FRS was first proposed by Dubois and Prade [4] and then studied in detail in [4, 17]. Interested readers could consult the reference [18] for more detailed summary of the development of FRS.

### 2.1 FRS

Fuzzy set theory, as one kind of generalization of set theory, is a mathematical tool for dealing with fuzziness and vagueness [19]. Let  $U$ , called the Universe, be a nonempty set with a finite number of objects, and  $A$  be a fuzzy subset on  $U$ , which is defined as a mapping  $A : U \rightarrow [0, 1]$ .  $\forall x \in U$ ,  $A(x)$  is a fuzzy membership degree of  $x$  belonging to fuzzy set  $A$ .

A Fuzzy Decision Table, shortly denoted by  $FD$ , is defined as a triple of  $(U, R, D)$ . Each object in  $U$  is described by a non-empty finite set of attributes, denoted by  $R \cup D$ ;  $R$  denotes the set of condition attributes and  $D$  denotes the set of decision attributes,  $R \cap D = \emptyset$ ; Each attribute  $r \in R$  corresponds to a  $U \rightarrow [0, 1]$  mapping, that is, each attribute is fuzzy not crisp. Each crisp attribute  $r \in D$  corresponds to a  $U \rightarrow V_r$  mapping, in which  $V_r$  is the value set of  $r$  over  $U$ .



In most practical applications, the condition attributes are fuzzy values but decision attributes are crisp values. This type of fuzzy decision table is thus acted as the platform to study the method of statistical FRS in this paper.

For every subset  $P$  of  $R$  and given a triangular norm  $T$ , a fuzzy similarity relation on attribute subset  $P$  is defined as  $P(\cdot, \cdot)$  satisfying, for every  $x, y, z \in U$ , (1) Reflexivity ( $P(x, x) = 1$ ), (2) Symmetry ( $P(x, y) = P(y, x)$ ), (3)  $T$ -transitivity ( $P(x, y) \geq T(P(x, z), P(z, y))$ ).

To illustrate how to calculate fuzzy similarity relation  $P(\cdot, \cdot)$ , we take  $T_L = \max\{0, x + y - 1\}$  as special case of triangular norm  $T$ . The similarity degree satisfying  $T_L$ -triangular norm can be calculated as follows,  $\forall r \in P, \forall x, y \in U$ ,  $P(x, y) = \min_{r \in P}(r(x, y))$ , where  $r(x, y) = 1 - \max(r(x), r(y)) + \min(r(x), r(y))$ .

The concept of FRS was first proposed by Dubois and Prade [4], their idea was as follows: Let  $U$  be a nonempty Universe and  $R$  a fuzzy binary relation on  $U$  which is a binary relation satisfying reflexivity and symmetry, A fuzzy rough set is an order pair  $(\underline{R}A, \overline{R}A)$  of a fuzzy set  $A$  on  $U$  such that for every  $x \in U$ ,

$$\underline{R}A(x) = \inf_{u \in U} \max\{1 - R(x, u), A(u)\}; \overline{R}A(x) = \sup_{u \in U} \min\{R(x, u), A(u)\}.$$

## 2.2 Attribute Reduction Based on FRS

Attribute reduction, as one kind of feature selection based on FRS, is able to select features that preserve the discernibility ability of original ones, but do not attempt to maximize the class separability [13]. To support efficient attribute reduction, many heuristic algorithms have been developed in FRS [3, 6, 7, 9, 18]. Here, we review one representative heuristic attribute reduction method: the reduction method based on dependency function [7, 9].

In the following some notations, such as positive region, dependency function and reduction of FRS, are given. In a fuzzy decision table  $FD = (U, R \cup D)$ , the positive region of  $x \in U$  relative to  $R$  is defined as  $POS_R(x) = \underline{R}([x]_D)(x)$ . Here,  $[x]_D = \{y \in U : D(x, y) = 1\}$  represents the set containing all the objects in  $U$  with the same decision classes to  $x$ . For any  $y \in U$ ,  $[x]_D(y) = \begin{cases} 1, & \text{if } D(x, y) = 1 \\ 0, & \text{if } D(x, y) = 0 \end{cases}$ . The dependency degree of  $D$  on  $R$ , denoted by  $Dep_R(D)$ , is defined as  $Dep_R(D) = \sum_{x \in U} POS_R(D)(x) / |U|$ .

The key idea of attribute reduction in RS of keeping dependency function invariant is also adopted to reduce attributes in FRS. In a fuzzy decision table  $FD = (U, R \cup D)$  and  $P \subseteq R$ ,  $P$  is called a reduct of  $R$  with respect to (w.r.t.)  $D$  if  $P$  satisfies the following two statements: (1)  $Dep_R(D) = Dep_P(D)$ ; (2) for any  $b \in P$ ,  $Dep_{P-\{b\}}(D) \neq Dep_R(D)$ . Actually, a reduct could be seen as one feature selection result which keeps the indiscernibility information invariant.

## 3 Random Sampling Based FRS

The topological explanation of the lower approximation of FRS is that  $\forall x \in U$ , its lower approximation is the minimum distance to the points from the different classes.

That is to say the computation complexity of the lower approximation is the square of all objects in the Universe. When the size of the data becomes large, the computation of lower approximation is enormous. Therefore, this paper gives up the method to compute the lower approximation on the whole universe, but randomly selecting some objects from the Universe, to form a sample set and then calculate the random approximation.

**Definition 1:** Given a Fuzzy Decision Table  $FD = (U, R, D)$  and  $N$  objects are randomly selected to form a sample  $S$ , then  $\forall x \in S$ , random lower and upper approximation which  $x$  belongs to  $[x]_D$  are defined as:

$$\begin{aligned} \underline{R}^S[x]_D(x) &= \inf_{u \in S} \max\{1 - R(x, u), [x]_D(u)\}; \quad \overline{R}^S[x]_D(x) \\ &= \sup_{u \in S} \min\{R(x, u), [x]_D(u)\}. \end{aligned}$$

To keep the distribution of the class labels unchanged, the proportional stratified sampling method is adopted. That is, the proportion of the class labels on the sample  $S$  is in line with that on the Universe.

Definition 1 provides an approximation method based on random sampling, the random approximation needs less computation. Based on the random approximation, the random positive region is defined as follows:

Given a Fuzzy Decision Table  $FD = (U, R, D)$  and  $N$  objects are randomly selected to form a sample  $S$ , then  $\forall x \in S$ , the random positive region of  $x$  satisfies  $POS_B^S(x) = \underline{R}^S[x]_D(x)$ ; the random dependence of  $P \subseteq R$  is defined as:  $\gamma_P^S = \frac{\sum_{x \in S} POS_P^S(x)}{|S|}$ .

Based on the above definitions, the random reduction can be defined as follows.

**Definition 2:** Given a decision table  $FD = (U, R, D)$  and, some samples are obtained by random sampling, i.e.,  $\{S_1, S_2, \dots, S_n\}$ ; Given attribute subset  $P \subseteq R$ , if  $P$  satisfies the following statements, we call  $P$  is a random reduction (in which  $\alpha \in [0, 1]$ ).

$$(1) \forall S_i \in S, |\gamma_R^{S_i} - \gamma_P^{S_i}| < \alpha; (2) \nexists b \in P, \forall S_i \in S, \left| \gamma_R^{S_i} - \gamma_{P - \{b\}}^{S_i} \right| < \alpha.$$

Definition 2 defines a minimal conditional attribute subset that make the dependence of the identification information on multiple samples invariant.

## 4 Random Reduction Algorithm

In the classical reduction algorithm, adding attributes in each iteration requires the calculation of lower approximation and positive region on the whole Universe, and then find one attribute makes the dependency degree increase maximally. If the scale of the Universe is large, the computation cost of each iteration is very high. To solve this problem, we propose a random reduction algorithm (RAR) based on random FRS, which is designed as follows:

---

 Algorithm 1 : Random reduction algorithm ( RAR )
 

---

 Input :  $DT = \langle U, R, D \rangle, N(\text{round number}), k(\text{sample number}), f_1(\text{sample qualified threshold})$ 

 Output :  $redu$ 
**LET**  $lef = R; redu = \phi; repeat = 0; \gamma_p = 0;$ 
**WHILE**  $repeat < N$  , **DO**

 In  $U$  use stratified sampling method select  $k$  samples randomly, as sample  $S_{cur}$ ;

 In  $S_{cur}$  find the attribute with maximum dependency increment  $a$  ,

 Write  $\gamma_c = \gamma_{redu|a}^{S_{cur}} = \max\{\gamma_{redu|a_i}^{S_{cur}}, a_i \in lef\}$ ;

**IF**  $\gamma_c \leq \gamma_p$  or  $(\gamma_c > \gamma_p$  and  $|\frac{\gamma_c - \gamma_p}{\gamma_c}| < f_1)$  , **THEN**

repeat + +;

**ELSE**
 $redu = redu \cup a;$ 
 $lef = lef - a;$ 
 $\gamma_p = \gamma_c;$ 
 $repeat = 0;$ 
**END IF**
**END WHILE**
**RETURN**  $redu;$ 


---

The design of algorithm 1 (RAR) is based on the method of maximum correlation and the minimum redundancy. Maximum correlation is achieved by looking for attributes with the greatest increment in dependency in each iteration. Minimum redundancy is achieved by selecting samples with the minimal overlapping information. In this algorithm,  $N$  is usually set as 10 or other larger integers. The larger  $N$  is, the more running time RAR needs.  $k$  is usually set according to  $t^2 \frac{P(1-P)}{d^2}$  where  $t$  is the bilateral  $1 - e$  quantile of standard normal distribution; and  $d \in [0, 1]$  is absolute error,  $p$  is set as 0.5.  $f_1$  is a sample qualified threshold, The larger  $f_1$  is, the harder is to find the satisfactory sample.

RAR are calculated only on the samples which greatly reduces the computation in each iteration. in RAR, the complexity of computing  $\gamma_B^U$  is  $O(|S|^2)$ . In CAR the complexity of computing  $\gamma_B^U$  is  $O(|U|^2)$ . We suppose the number of iteration in RAR is  $M$ , the number of iteration in CAR is  $N$ . Then the time complexity of CAR is  $O(N|R||U|^2)$ , the time complexity of RAR is  $O(M|R||S|^2)$ . Thus, RAR has the obvious advantages from the aspect of scalability.

## 5 Numerical Experiment

In this section, we conduct some numerical experiments. The experimental environment is set as follows: (a) CPU: Intel(R) Xeon(R) CPU E7-4820 v2 @ 2.00 GHz; (b) Memory: 500G; (c) Programing language: C++; (d) Operating system: Linux.

The datasets used in the experiments are derived from the UCI database [8], and the details of which are prescribed in Table 1.

**Table 1.** The description of selected datasets

Datasets	Obj. No.	Attr. No.	Class
Letter	20000	16	26
Gas	13910	128	6
Credit	30000	23	2
Magic	19020	10	2
Sat	4435	36	2
Spam	4601	57	2
Waveform	5000	21	3

### 5.1 Compare CAR with RAR

This subsection experimentally compares the speed of the classical reduction algorithm (CAR) and Random Reduction Algorithm (RAR). The detailed experimental results are shown in Table 2.

**Table 2.** The comparison of the execution time between CAR and RAR

(1) The datasets whose size is larger than 10000							
Dataset	Data Size	Attribute No.	The execution time of CAR		The execution time of RAR		Ratio CAR/RAR
Magic	19020	10	2338.29 s	(=38 m 58 s)	99 s	(=1 m 39 s)	<b>23.62</b>
Letter	20000	16	9917.30 s	(=2 h 45 m 17 s)	391.04 s	(=6 m 31 s)	<b>25.36</b>
Credit	30000	23	25616.88 s	(=7 h 6 m 57 s)	345.54 s	(=5 m 46 s)	<b>74.14</b>
Gas	13910	128	19482.96 s	(=5 h 24 m 43 s)	5526.47 s	(=1 h 32 m 6 s)	3.53
Average	20732.5	44.3	14338.86 s	(=3 h 58 m 59 s)	1590.51 s	(=26 m 31 s)	31.66
(2) The datasets whose size is less than 10000							
Dataset	Data Size	Attribute No.	The execution time of CAR		The execution time of RAR		Ratio CAR/RAR
Sat	4435	36	2846.45 s	(=47 m 26 s)	1041.02 s	(=17 m 21 s)	2.73
Spam	4601	57	346.42 s	(=5 m 46 s)	1004.56 s	(=16 m 45 s)	0.34
Waveform	5000	21	1804.88 s	(=30 m 5 s)	524.17 s	(=8 m 44 s)	3.44
Average	4678.67	38	1665.92 s	(=27 m 46 s)	856.58 s	(=14 m 17 s)	2.17

Several observations can be drawn from Table 2. First, Table 2 shows RAR is significantly or even dramatically faster than that of CAR on the datasets with large number of objects. For example, the average execution time of CAR on the dataset whose size larger than 10000 is 3 h 58 min, whereas that of RAR is just 26 min 31 s. And the average ratio of CAR/RAR is 31.66. This is because random sampling significantly reduces the calculation when the data size is large.

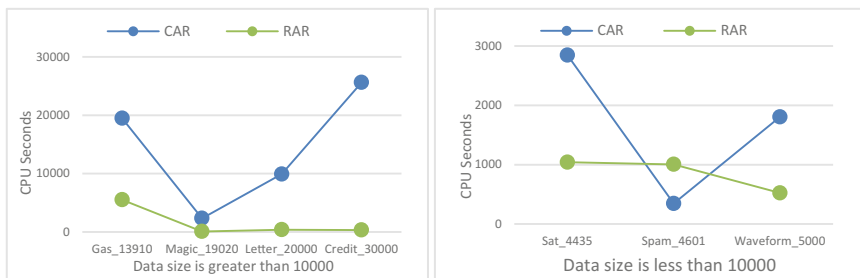
Also, Table 3 shows that RAR sometimes works slightly better than CAR on the datasets which size is less than 10000. For example, the average ratio of CAR/RAR is just 2.17.

**Table 3.** The comparison of the execution time between IAR and RAR

(1) The datasets whose size is larger than 10000							
Dataset	Data Size	Attr. No.	The exec. time of IAR		The exec. time of RAR		Ratio IAR/RAR
Magic	19020	10	503.65 s	(=8 m 24 s)	99 s	(=1 m 39 s)	5.09
Letter	20000	16	1371.70 s	(=22 m 52 s)	391.04 s	(=6 m 31 s)	3.51
Credit	30000	23	3993.82 s	(=1 h 6 m 34 s)	345.54 s	(=5 m 46 s)	11.56
Gas	13910	128	2720.55 s	(=45 m 21 s)	5526.47 s	(=1 h 32 m 6 s)	0.49
Average	20732.5	44.3	2147.43 s	(=35 m 47 s)	1590.51 s	(=26 m 31 s)	5.16
(2) The datasets whose size is less than 10000							
Dataset	Data Size	Attr. No.	The exec. time of IAR		The exec. time of RAR		Ratio IAR/RAR
Sat	4435	36	137.36 s	(=2 m 17 s)	1041.02 s	(=17 m 21 s)	0.13
Spam	4601	57	39.79 s	(=40 s)	1004.56 s	(=16 m 45 s)	0.04
Waveform	5000	21	169.30 s	(=2 m 49 s)	524.17	(=8 m 44 s)	0.32
Average	4678.67	38	115.48 s	(=1 m 55 s)	856.58 s	(=14 m 17 s)	0.16

To further compare the execution time between CAR and RAR, we plot the execution time changing with the data size, see Fig. 1.

Some observations can be drawn as follows. Figure 1 shows that the trend of RAR does not change obviously with the increasing data size on most cases. This shows that the execution time of RAR is relatively stable no matter how large the dataset is. This is because the execution time of RAR is closely related with the size of randomly sample, but less related with the size of the whole datasets. It is easy to see that in Fig. 1, RAR is not stable on the data set ‘Sat’, ‘Spam’ and ‘waveform’. This is because the data sizes of ‘Sat’ and ‘Spam’ are too small and then the randomly sampled subset cannot statistically represent the whole dataset. As a result, we draw a conclusion that RAR is not suitable for the small-scale datasets, RAR is more efficient for large-scale data.

**Fig. 1.** The trends of execution time according to the data size

## 5.2 Compare IAR with RAR

This subsection experimentally compares the Incremental Reduction Algorithm (IAR) [21] and Random Reduction Algorithm (RAR).

This incremental reduction algorithm is an algorithm which split the datasets into several parts, and then update the reduct obtained on one part with the help of other parts, which is a little similar with the random reduction algorithm. As a result, we compare IAR with RAR. The detailed experimental results are shown in Table 3. Here, the whole datasets are split into 5 parts. One part is seen as the original part, the others are seen as the successively arriving data. The execution time of IAR is the sum of IAR working these parts.

Several observations can be drawn from Table 3. First, Table 3 shows that RAR works faster than IAR on the datasets with the large number of datasets. RAR runs a little faster than IAR on most datasets, whose size is larger than 10000. Whereas, RAR works slower than IAR on the datasets whose data size is less than 10000. For example, in Table 3(2) the average execution times of IAR and RAR on the datasets whose data size is less than 10000 are 1 min 55 s and 14 min 17 s, respectively. This is because RAR is not stable on small datasets, especially on Sat and Spam whose data sizes are less than 5000. The above observations show that is RAR is not suitable for the small-scale datasets, but efficient for large-scale data.

## 6 Conclusions

A fuzzy rough based feature selection algorithm by using random sampling technique is proposed in this paper. The main contribution of this paper is the introduction of the idea of random sampling in the selection of features based on minimum redundancy and maximum relativity. This paper is an interesting attempt which combine the random sampling and fuzzy rough reduction. In the near future, we would like to induce some useful fuzzy rough rules by using random sampling techniques.

## References

1. Bluma, A.L., Langley, P.: Selection of relevant features and examples in machine learning. *Artif. Intell.* **97**, 245–271 (1997)
2. Chen, H.M., Li, T.R., Ruan, D., Lin, J.H., Hu, C.X.: A rough-set based incremental approach for updating approximations under dynamic maintenance, environments. *IEEE Trans. Knowl. Data Eng.* **25**, 274–284 (2013)
3. Chen, D.G., Wang, X.Z., Yeung, D.S., Tsang, E.C.C.: Rough approximations on a complete completely distributive lattice with applications to generalized rough sets. *Inf. Sci.* **176**, 1829–1848 (2006)
4. Dubois, D., Prade, H.: Rough fuzzy sets and fuzzy rough sets. *Int. J. Gen. Syst.* **17**, 191–208 (1990)
5. Hnich, B., Rossi, R., Tarim, S.A., Prestwich, S.: Filtering algorithms for global chance constraints. *Artif. Intell.* **189**, 69–94 (2012)
6. Hu, Q.H., Zhang, L., An, S., Zhang, D., Yu, D.R.: On robust fuzzy rough set models. *IEEE Trans. Fuzzy Syst.* **20**(4), 636–651 (2012)
7. Hu, Q.H., Yu, D.R., Xie, Z.X.: Information-preserving hybrid data reduction based on fuzzy-rough techniques. *Pattern Recogn. Lett.* **27**, 414–423 (2006)
8. <http://www.ics.uci.edu/~mllearn/MLRepository.html>

9. Jensen, R., Shen, Q.: Fuzzy-rough attributes reduction with application to web categorization. *Fuzzy Sets Syst.* **141**, 469–485 (2004)
10. Joshi, S., Jermaine, C.: Materialized sample views for database approximation. *IEEE Trans. Knowl. Data Eng.* **20**(3), 337–351 (2008)
11. Karabadjji, N.E., Seridi, H., Khelif, I., Azizi, N., Boulkroune, R.: Improved decision tree construction based on attribute selection and data sampling for fault diagnosis in rotating machines. *Eng. Appl. Artif. Intell.* **35**, 71–83 (2014)
12. Léon, B., Olivier, B.: The tradeoffs of large scale learning. In: *Advances in Neural Information Processing Systems*, pp. 161–168 (2008)
13. Liang, J.Y., Wang, F., Dang, C.Y., Qian, Y.H.: A group incremental approach to feature selection applying rough set technique. *IEEE Trans. Knowl. Data Eng.* **26**(2), 294–308 (2014)
14. Motwani, R., Raghavan, P.: *Randomized Algorithms*. Cambridge University Press, Cambridge (1995)
15. Provost, F., Jensen, D., Oates, T.: Efficient progressive sampling. In: *Proceedings of KDD 1999*, pp. 23–32 (1999)
16. Qian, Y.H., Liang, J.Y., Pedryc, W., Dang, C.Y.: Positive approximation: an accelerator for attribute reduction in rough set theory. *Artif. Intell.* **174**, 597–618 (2010)
17. Tarim, S.A., Manandhar, S., Walsh, T.: Stochastic constraint programming: ascenario-based approach. *Constraints* **11**, 53–80 (2006)
18. Dai, J., Hu, H., Wu, W.Z., et al.: Maximal discernibility pairs based approach to attribute reduction in fuzzy rough sets. *IEEE Trans. Fuzzy Syst.* **PP**(99), 1 (2017)
19. Zadeh, L.A.: Fuzzy sets. *Inf. Control* **8**, 338–353 (1965)
20. Cover, T., Hart, P.: Nearest neighbor pattern classification. *IEEE Trans. Inf. Theory* **13**(1), 21–27 (1967)
21. Liu, Y., Zhao, S., Chen, H., Li, C., Lu, Y.: Fuzzy rough incremental attribute reduction applying dependency measures. In: Chen, L., Jensen, C.S., Shahabi, C., Yang, X., Lian, X. (eds.) *APWeb-WAIM 2017*. LNCS, vol. 10366, pp. 484–492. Springer, Cham (2017). [https://doi.org/10.1007/978-3-319-63579-8\\_37](https://doi.org/10.1007/978-3-319-63579-8_37)
22. Zhang, L., Suganthan, P.N.: A survey of randomized algorithms for training neural networks. *Inf. Sci.* **364**, 146–155 (2016)
23. Ott, R.L., Longnecker, M.T.: *An introduction to statistical methods and data analysis*. Nelson Education (2015)
24. Wen, X., Shao, L., Xue, Y., Fang, W.: A rapid learning algorithm for vehicle classification. *Inf. Sci.* **295**, 395–406 (2015)
25. Anagnostopoulos, E., Emiris, I.Z., Psarros, I.: Randomized embeddings with slack, and high-dimensional Approximate Nearest Neighbor. *Comput. Sci.* (2016)
26. Pawlak, Z.: Rough sets. *International Journal of Computer and Information Science* **11**, 341–356 (1982)
27. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. arXiv preprint [arXiv: 1412.6980](https://arxiv.org/abs/1412.6980) (2014)
28. Thisted, R.A.: *Elements of statistical computing: Numerical computation*. Routledge, New York (2017)



# Segmenting Sound Waves to Support Phonocardiogram Analysis: The PCGseg Approach

Hajar Alhijailan<sup>1,2</sup>(✉), Frans Coenen<sup>3</sup>, Jo Dukes-McEwan<sup>4</sup>,  
and Jeyarajan Thiyagalingam<sup>5</sup>

<sup>1</sup> Department of Computer Science, The University of Liverpool, Liverpool, UK  
h.alhijailan@liverpool.ac.uk

<sup>2</sup> College of Computer and Information Sciences, King Saud University, Riyadh,  
Saudi Arabia

<sup>3</sup> Department of Computer Science,  
The University of Liverpool, Liverpool, UK  
coenen@liverpool.ac.uk

<sup>4</sup> Small Animal Teaching Hospital, The University of Liverpool,  
Leahurst Campus, Neston, UK  
jdmcewan@liverpool.ac.uk

<sup>5</sup> Department of Electronics and Electrical Engineering,  
The University of Liverpool, Liverpool, UK  
tjeyan@liverpool.ac.uk

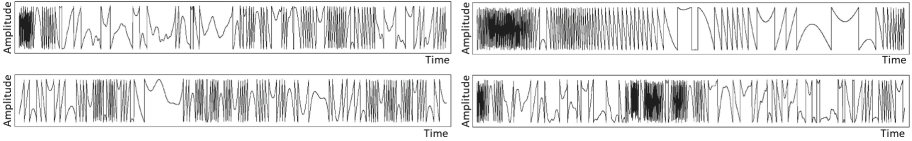
**Abstract.** The classification of Phonocardiogram (PCG) time series, which is often used to indicate the heart conditions through a high-fidelity sound recording, is an important aspect in diagnosing heart-related medical conditions, particularly on canines. Both the size of the PCG time series and the irregularities featured within them render this classification process very challenging. In classifying PCG time series, motif-based approaches are considered to be very viable approach. The central idea behind motif-based approaches is to identify reoccurring subsequences (which are referred to as motifs) to build a classification model. However, this approach becomes challenging with large time series where the resource requirements for adopting motif-based approaches are very intensive. This paper proposes a novel two-layer PCG segmentation technique, called as PCGseg, that reduces the overall size of the time series, thus reducing the required for generating motifs. The evaluation results are encouraging and shows that the proposed approach reduces the generation time by a factor of six, without adversely affecting classification accuracy.

## 1 Introduction

Time series analysis is concerned with the extraction of important parameters from a given time series, which is an extension of point series data [1]. The distinguishing feature of such data is that it comprises an ordered sequence



of values. Although there a whole class of problems associated with time series analysis, the work presented in this paper is directed towards the classification of canine Phonocardiogram (PCG) time series. Some example time series fragments are given in Fig. 1. Within PCG time series, certain recurring sub-sequences, known as motifs, characterize certain heart conditions among canines. As such, identifying these motifs is a basic requirement prior to more detailed diagnosis.



**Fig. 1.** PCG time series examples

A number of motif discovery techniques have been proposed in the literature [2–5]. However, in the context of PCG data, two significant issues are: (i) the time to generate/identify these motifs and (ii) the accuracy of resulting classification. The first problem is often exacerbated by the size of PCG time series. The second is concerned with the quality of the identifiable motifs, so that they are appropriate enough to be reasonable differentiators of different classes. These two issues are inter-linked, albeit being data dependent. And hence, if motif generation can be made more efficient, additional resource will be available to select “better” motifs given a time constraint.

A suggested solution to the first, and consequently provide for the second, is to pre-process the time series so as to reduce their size in such a way that salient features are preserved [6]. One common pre-processing technique is time series segmentation [6]. The basic process is to divide the time series into blocks. The intended advantage is that the resulting segmented time series is much shorter than the original while preserving necessary salient features. From the literature a variety of algorithms have been proposed to obtain a good high level segmentation of time series data [6–12]. An extensive coverage of the literature is provided in Sect. 2.

Amongst all these techniques, the most appropriate segmentation mechanism is often problem-specific and dependent on the application domain under consideration. The application domain at which the work presented in this paper is the classification of canine PCGs according to a variety of heart conditions that can be identified from PCG data. A PCG is a two dimensional time series, where the values are amplitudes. The average number of points (length) in the collected time series was over 355,000; a significant number and too many to be processed easily using established motif generation mechanisms. As mentioned before, some form of pre-processing, segmentation, to reduce the overall file size was thus considered to be appropriate. By adopting this approach, using some form of segmentation, the intention is to reduce the size of the collected time series so that a useful classification model can be generated in an efficient manner.

The main contribution is thus a bespoke, two layer, PCG time series segmentation mechanism, PCGseg, which significantly reduces the processing time (by a factor of six) without adversely affecting the resulting accuracy. The approach is fully described and evaluated using a realistic motif-based classification scenario.

The rest of this paper is organized as follows. Section 2 gives a review of the previous work specific to the application domain. A formalism for the problem domain is given in Sect. 3. The proposed segmenting method, which is specifically designed for PCG data recorded in WAVE file format is presented in Sect. 4. Section 5 presents the evaluation of the proposed PCGseg approach. Some discussion is then presented in Sect. 6. This paper is concluded with some summarising remarks and directions for future work in Sect. 7.

## 2 Previous Work

This section provides a review of existing work concerning the segmentation of time (point) series. The section is divided into two subsections. Subsect. 2.1 considers previous work concerning the segmenting of point series, whilst Subsect. 2.2 examines previous work on PCG segmentation.

### 2.1 Segmenting Point Series

As noted in the introduction to this paper, segmentation is the process of dividing a whole entity into constituent or distinct elements, the term is frequently used in the context of image analysis [13]. Techniques for achieving effective and efficient segmentation remain an area of current research. The main objective of segmentation is to reduce the amount of data so that the features of interest are retained. In the context of time series, a number of mechanisms have been used to achieve segmentation, these include: (i) Fourier Transforms [9], (ii) Wavelets [10], (iii) Symbolic Mappings [8] and (iv) Piecewise Linear Representation (PLR) [7, 12]. PLR is the most common of these four mechanisms. It is also of relevance with respect to the work presented here because it is achieved in a similar manner (using a moving window approach). The fundamental idea of PLR is to translate a given time series  $T$  into a model  $\bar{T}$  which comprises a number of “best fitting” straight lines (segments). However, the nature of PCG curves (point series), see Fig. 1, is such that PLR is not appropriate for the PCG application considered in this paper. Line fitting, whatever form this might take, is not sufficiently descriptive for the purpose of PCG classification.

Regardless of the segmentation approach used, each can be implemented using one of three basic mechanisms as follows [6]: Sliding Window, Top Down and Bottom Up. The first approach is faster than the other two. However, decisions are made without any deep exploration of the time series as in the case of the later two approaches; this in turn may affect the quality of the segmentation. Both Top Down and Bottom Up give good results, but they tend to be impractical given large data sets as they require a scan of the entire time series. The method proposed in this paper uses the Sliding Window mechanism, however

the segmentation is not related to similarity with the underlying point series but with how accurately the segments represent the “shape” of the underlying point series sub-sequence.

## 2.2 PCG Segmentation

As noted above, segmentation is the term used to subdivide a signal trace into sub-sequences according to some criteria. A PCG trace can be divided into single heart (cardiac) cycles each comprised of four principal components: (i) first cardiac sound (S1), (ii) *systole*, (iii) second cardiac sound (S2) and (iv) *diastole*. The Extraction of PCG segments from several cardiac cycles (heart beats) is usually achieved with help of an ECG signal recorded at the same time and/or a Carotid Pulse (CP) which can then be used as a reference signal [14,15]. In the case of the work considered in this paper, no such reference signals are available.

There is some reported research directed at the segmentation of PCGs without a reference signal [16–19] but with some limitations. Some of this research [17] concentrates mainly on finding first and second heart sounds ( $S_1$  and  $S_2$ ) in a “wveshape”, which is a reflection of the PCG as a form of five visible deflections per each heart beat. Other work, such as [18], is directed at extracting heart events ( $S_1$ ,  $S_2$ ,  $S_3$  and  $S_4$ ) in a spectrogram. However, finding heart sounds (events) in a few seconds of a recorded PCG signal is not suited to Time Series Motif Discovery; a much longer time series is required with respect to canine PCG data, the target domain used with respect to the evaluation presented later in this paper. In [20] it was observed that even a 10 s series, featuring 12 to 27 beats for a normal resting dog, is not adequate for motif discovery. In addition, the focus of the majority of the existing work on PCG segmentation is directed at heart event detection in normal cardiac activity using the energy of the signal which in turn is affected by noise [19] or murmurs [16]; whereas this paper aims at classifying regular and irregular cardiac activities.

## 3 Formalism

A time (point) series  $P$  comprises a sequence of  $n$  data values  $\{p_1, p_2, \dots, p_n\}$ . Using PCGseg a segmented point series  $S$  consists of a sequence of  $m$  segments  $\{S_1, S_2, \dots, S_m\}$  where each segment  $S_i$  represents some sub-sequence of  $P$ . Each segment  $S_i$  is defined in terms of a tuple of the form  $\langle S_p, S_C \rangle$ , where  $S_p$  is the parent segment and  $S_C$  is a set of constituent sub-segments  $\{S_{c_1}, S_{c_2}, \dots\}$ .  $S_P$  is defined in terms of a tuple of the form:  $\langle shape, type, length \rangle$ , where: *shape* is the nature of the point series defined by the segment,  $\{slant, vertical, dome, flat\}$ ; *type* is the direction of the shape, either *up* or *down*; and *length* is the number of points represented by the segment. More specifically the length of a segment is the difference between the start and end index values. Thus a shape comprised of two points will have length 1 and so on. Each element  $S_{c_i} \in S_C$  is represented by a second tuple of the form:  $\langle type, length, depth \rangle$ , where: *type* is the nature of the point series defined by the sub-segment  $\{up, down, flat\}$ , *length* is the

length of the sub-segment (calculated as described above); and *depth* is the difference between the maximum and minimum amplitude values represented by the sub-segment in question. Given this representation a motif  $M$  is then some subset of  $S$  ( $M \subset S$ ) that repeats and is thus deemed to be representative of the underlying point series.

A collection of  $x$  segmented, and labeled, point series is given by  $\mathbf{D} = \{D_1, D_2, \dots, D_x\}$ , where each  $D_i$  is a tuple of the form  $\langle S_i, c_i \rangle$  where  $S_i$  is a segmented point series and  $c_i$  is a class label taken from a set of class labels  $C$ . The aim is to identify a collection of motifs  $\mathbf{L} = \{L_1, L_2, \dots\}$  such that each  $L_i$  is a tuple of the form  $\langle M_i, c_i \rangle$  where  $M_i$  is a motif and  $c_i$  is a class label. The set  $\mathbf{L}$  can then be used to build a classification model of some kind.

## 4 PCGseg

The proposed PCGseg algorithm is underpinned by the idea of capturing the “shapes” that exist in a PCG sequence. The fundamental idea is that a PCG time sequence can be conceptualised in terms a series of shapes and sub-shapes (segments and sub-segments). From Fig. 1 four distinct shapes can be identified: (i) slant, (ii) vertical, (iii) dome and (iv) flat. It is also important to note that the vertical shape always occurs between any two other shapes. In other words it can be regarded as a separator; this feature is used in the context of the proposed PCGseg mechanism to identify the start and end points of segments.

As already noted in Sect. 3, a segment is defined by a tuple of the form  $\langle shape, type, length \rangle$ , and as also already noted above, the possible values for the *shape* variable are:  $\{vertical, slant, dome, flat\}$ . Each segment is defined in terms of a conceptual Minimum Bounding Box (MBB) surrounding it. The x-dimension of the MBB of a segment corresponds to the value of the *length* variable associated with the segment. More specifically, each shape is defined as follows:

1. **slant:** A slant shape comprises a sequence of three or more points ( $length \geq 2$ ) such that the start and end points are at opposite corners of the MBB and the difference between the start and end point amplitude values is greater than a threshold  $t2$ .
2. **vertical:** A vertical shape is a special case of the slant shape whose *length* is 1. This shape appears very often, always between two other shapes; it can thus be viewed as a separator.
3. **dome:** A dome shape comprises a sequence of three or more points ( $length \geq 2$ ) such that the start and end points are on the same side (top or bottom) of the associated MBB. This is defined in terms of the difference between the start and end point amplitude values which must be less than a threshold  $t2$ .
4. **flat:** A flat shape is a special case of the dome shape, comprised of two or more points ( $length \geq 1$ ) and whose depth (maximum difference between amplitudes) is less than a predefined threshold  $t3$ .

The *type* of a shape is determined by the “direction” of the shape. The possible values for the *type* variable are  $\{up, down\}$ . The value for the *type* variable is defined by the first two points in the sequence. If the amplitude of the second point is greater than the first, the type is *up*. If it is less than the first, the type is *down*. Where the first two points have the same amplitude value, a rare occurrence, this is dealt with by considering their location within the overall time series. If both amplitudes are below the average value, a threshold  $t_4$ , they are considered to have the type *up* and otherwise, the type is *down*.

Whatever the case, a sub-segment, as noted in Sect. 3, is described by a tuple of the form  $(type, length, depth)$ . The possible values for the type variable are:  $\{up, down, flat\}$ . Note that the values for the *type* variable associated with a sub-segment are not the same as those associated with a segment. The value for the *type* variable is defined by all points in the sub-segment. If the amplitude of all points is increasing, the type is *up*. If it is decreasing, the type is *down*. Otherwise, the type is *flat*.

From the foregoing, we have a set of five thresholds,  $\{t_1, t_2, t_3, t_4, t_5\}$ , which are used to segment point series. They are particularly used for detecting vertical shapes, slant shapes, dome shapes, dome and flat shape types and sub-segment types respectively.

The motivation for the proposed PCGseg mechanism was to represent time series, and PCG point series in particular, in terms of their constituent shapes and sub-shapes in a two level hierarchy instead of averaging values taken periodically [21] or extracting trends [22]. The conjectured advantages were that:

1. The main information, which would be lost in the case of the application of the averaging method, is preserved. The significance is that the rate of change in PCG records is quite high. Averaging may still work if a very narrow window is used however this would defeat the objective of the segmentation, to reduce the data volume.
2. A more succinct segmentation would be produced, than that produced by earlier segmentation mechanisms, by considering “parent” and “child” shapes (segments and sub-segments) where the child shapes represent “trends” in the parent shape. Recall that a parent shape can have any number of sub-shapes (trends); this will be the case where the parent shape features many irregularities and fluctuations.
3. Prediction using point series requires a substantial amount of matching of point series sub-sequence. The proposed two-level hierarchical segmentation allows for “early abandonment” where the parent segment does not fit the comparator segment (no need to go down to the next level).

#### 4.1 Motif Detection

Once the entire data set had been segmented, motifs can be identified. The idea was to store the identified motifs, together with an associated class label, in a “bank” of motifs which could then be used to classify (label) previously unseen PCG records. The adopted approach was founded on the MK motif discovery

algorithm [23], a well established motif detection algorithm that operates by identifying and testing a number of candidate motifs and selecting the first  $n$  where matches are found. The MK algorithm operates using a window of size  $\omega$  (measured in terms of a number of points) and a reference value  $r$ . A sequence of  $r$  random windows are generated, of length  $\omega$ , and a best similarity value is obtained for each by comparing it with all other sub-sequences of length  $\omega$  in the given point series. The top  $n$  are then selected. In [23],  $n = 2$  was used, this value was also used with respect to the evaluation reported on later in this paper.

The original version of the MK algorithm, as described in [23], was designed to operate with point series, and used Euclidean Distance as a measure of similarity. However, this measure would clearly be unsuitable in the case of segmented data and thus an alternative distance function was required so that distances between motifs comprised of segments could be obtained. To measure how well two potential motifs match, five criteria were considered, in turn, in such a way that a policy of early abandonment could be adopted. The first criterion is the number of (parent) segments. The second is segment shapes and types. Then, the number of (children) sub-segments followed by sub-segment types. Lastly, average sub-segment lengths and depths. Thus when comparing two motifs, if any one of the first four criteria was not satisfied, the comparison would be abandoned without further comparison being required. In more detail, both sets of parent segments must be the same shape and type, in the same order. The number of sub-segments that feature in each segment should be identical, as should the order, shapes and types of the sub-segments.

Lastly, the fifth criterion, the Root Mean-Square Distance (RMSD) over the lengths and depths of the sub-segments was calculated and a similarity index,  $sim$ , was produced using (1) where  $X_{c_i}$  and  $Y_{c_i}$  are sub-segments in the two motifs  $X$  and  $Y$  being compared and  $j$  is the number of sub-segments.

$$sim = \frac{\sqrt{\sum_{i=1}^j (length_{X_{c_i}} - length_{Y_{c_i}})^2} + \sqrt{\sum_{i=1}^j (depth_{X_{c_i}} - depth_{Y_{c_i}})^2}}{j}. \quad (1)$$

The strategy of using five criteria is obviously very strict. This was deemed appropriate for motif discovery. However, unsuitable for measuring the similarity between the test and training data in the classification stage. In early experiments (not reported here), it was found that when using this strict strategy, the vast majority (74%) of the test data was not being classified at all because of the strictness of the matching. Therefore, an alternative, more tolerant, approach was adopted for the classification stage when comparing a motif  $M1$  identified in a record to be classified and a motif  $M2$  in the bank of labelled motifs extracted from the training data (details not included here for reasons of space limitation).

## 5 Evaluation

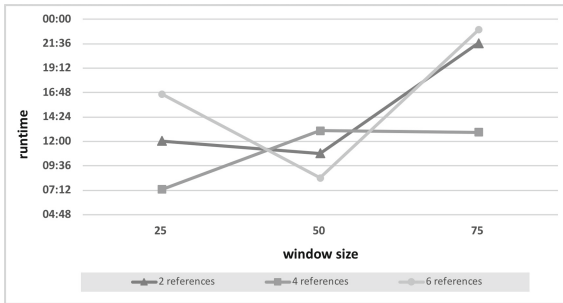
This section presents the evaluation of the proposed PCGseg approach. Recall that the motivation for the proposed approach was to speed up the classifier generation process without loss of accuracy. The operation of the proposed PCGseg approach was compared with the use of unsegmented data in terms of a PCG multi-class classification problem (the data set used is described in Subsect. 5.1). In both cases, the MK algorithm [23], presented earlier, was used to generate motifs. The evaluation was conducted in terms of accuracy and runtime; runtime to establish whether the PCGseg approach was faster or not, and accuracy to determine whether adoption of the PCGseg approach had a negative effect on accuracy or not. Recall also that the MK algorithm operates using two parameters: (i) the desired window size  $\omega$  and (ii) the number of candidates to be generated,  $r$ , known as the *reference value*. Using the PCGseg approach, the value for  $\omega$  was defined in terms of a number of segments, a range of values for  $\omega$  was considered  $\{25, 50, 75\}$  with respect to the evaluation reported here. With respect to the comparator approach, using unsegmented data,  $\omega$  was defined in terms of a number of points, a range of values for  $\omega$  was considered  $\{25000, 50000, 100000\}$ , selected so as to achieve broad equivalents with the selected number of segment  $\omega$  values. In both cases, a range of values for  $r$  was also considered  $\{2, 4, 6\}$ ; although it should be noted that in [23] it was reported that the value of  $r$  was not critical and that any value greater than five makes little difference. The three values for both  $\omega$  and  $r$  thus combined to give nine combinations, hence nine sets of experiments. As suggested in [23], the top two best motifs were retained for each record.

### 5.1 Evaluation Data Set

The data set used for the evaluation was a set of canine Mitral Valve disease Phonocardiograms (PCGs), encapsulated as WAVE files and (for the purpose of the evaluation) interpreted as time series such that the y-axis comprised *amplitude* values. The PCG data was collected, by staff at the University of Liverpool Small Animal Teaching Hospital, using electronic stethoscope equipment. In some cases, the recordings were done in stereo, in others in mono. In the case of the stereo WAVE files, two point series were extracted, one for each channel. This resulted was a 72 point series dataset. Each point series had a class label associated with it selected from the class attribute set  $\{B_1, B_2, C, Control\}$ . The first three class attributes are stages of Mitral Valve disease that appear in the collected data as defined by the European College of Veterinary Internal Medicine (ECVIM) [24]. The last class attribute represents PCGs that did not feature any disease (used for control purposes). The average length of a single point series was 355,484 points. Once PCGseg had been applied, it was reduced to 83,569 segments. A reduction in size by a factor of 4.25.

## 5.2 Runtime Evaluation

The runtime results obtained using the proposed PCGseg approach coupled with the MK algorithm are presented in Fig. 2. The runtimes were considerably shorter than those obtained using the MK algorithm applied to time series without segmentation. With segmentation, the average runtime for each experiment was about 14 h, without segmentation (results not included here for reasons of space limitation) the average runtime per experiment was some 90 h (more than six times greater than when using segmented data). From Fig. 2, it can be seen that when using  $r = 2$  and  $r = 6$  similar runtime patterns are produced, whereas when using  $r = 4$ , the behaviour is different. For  $r = 2$  and  $r = 6$ , the best runtime was obtained using a window size of  $\omega = 50$ ; whilst for  $r = 4$  using a window size of  $\omega = 25$  produced the best runtime.



**Fig. 2.** Runtime plots for Motif Generation using the MK algorithm applied to segmented PCG time series

## 5.3 Classification Accuracy

For the experiments to compare the classifiers accuracy, two different classification models were used: (i) the well known  $k$  Nearest Neighbour (KNN) classification model [25, 26] and (ii) Smallest Average Classification (SAC), a variation of KNN developed by the authors. KNN classification operates by finding the  $k$  most similar existing (labelled) records to a previously unseen record to be classified. For the experiments reported here,  $k = 1$  was used; thus the class associated with the most similar record was assigned to the record to be classified. KNN classification was chosen because it is frequently used in point series analysis [27, 28]. The SAC model calculates the similarity (average distance) between a new record to be classified and all previously stored records for each class; the new record will then be classified with the class label associated with the most similar class. Similarity was calculated using the approach presented at the end of Sect. 4 above.

Classification accuracy was measured in terms of Accuracy, Precision, Recall and F-Score; metrics all commonly utilised to evaluate classification models [29].



The evaluation was conducted using Ten Cross Validation (TCV) throughout. The average results obtained are presented in Tables 1 and 2, best results associated with each  $\omega$  value highlighted in bold font. Table 1 gives the results obtained using the proposed PCGseg approach (and both KNN and SAC), whilst Table 2 presents the results obtained using the raw, unsegmented data (and both KNN and SAC). From the tables, it can be seen, firstly, that the performance using segmented and unsegmented data was similar, a recorded best and worst accuracy using segmented data of 70.8% and 54.7%, and without segmentation a best and worst accuracy of 71.9% and 57.6%. In terms of classification model, there was also a very little difference in operation between the two. In terms of the  $\omega$  parameter, an argument can be made that, when using PCGseg, a value of  $\omega = 50$  was the most appropriate. In terms of the experiments using unsegmented data,  $\omega = 50000$  produced the best results. The chosen value for  $r$  seemed to have little effect, conforming the observation made in [23].

**Table 1.** Classification performance using segmented data and both KNN and SAC classification

$\omega$	$r$	KNN				SAC			
		Accuracy	Precision	Recall	F-Score	Accuracy	Precision	Recall	F-Score
25	2	<b>0.631</b>	<b>0.072</b>	<b>0.262</b>	<b>0.112</b>	0.596	0.088	0.250	0.118
	4	0.581	0.040	0.250	0.068	0.666	<b>0.120</b>	0.212	0.147
	6	0.596	0.047	0.250	0.077	<b>0.673</b>	<b>0.120</b>	<b>0.254</b>	<b>0.155</b>
50	2	<b>0.701</b>	0.100	0.250	0.142	0.701	0.104	0.250	0.145
	4	0.629	0.097	0.170	0.120	0.604	0.110	0.166	0.131
	6	<b>0.701</b>	<b>0.123</b>	<b>0.262</b>	<b>0.161</b>	<b>0.708</b>	<b>0.152</b>	<b>0.278</b>	<b>0.191</b>
75	2	<b>0.701</b>	0.100	0.250	0.142	0.547	0.065	0.120	0.069
	4	0.673	0.172	0.262	0.206	0.562	0.075	0.183	0.099
	6	0.681	<b>0.191</b>	<b>0.299</b>	<b>0.218</b>	<b>0.639</b>	<b>0.173</b>	<b>0.312</b>	<b>0.212</b>

## 6 Discussion

Looking at the results presented in Tables 1 and 2 in more detail, a great deal of variability can be discerned. This variability of the results was attributed to the fact that the MK algorithm selects motifs in a random manner and this randomness probably does not work well with high data volumes (segmented or otherwise). It is conjectured that the limited performance effectiveness, as reported above, was as a consequence of the random manner that candidate motifs were chosen when using the MK algorithm, which may in turn have led to the selection of motifs that were not especially indicative of a class. The MK algorithm chooses the best two similar sub-sequences to be motifs whilst not taking into consideration whether these motifs are in fact good indicators of class

**Table 2.** Classification performance using unsegmented data and both KNN and SAC classification

$\omega$	$r$	KNN				SAC			
		Accuracy	Precision	Recall	F-Score	Accuracy	Precision	Recall	F-Score
25000	2	<b>0.711</b>	0.105	0.250	0.146	<b>0.686</b>	0.094	<b>0.241</b>	0.133
	4	0.694	<b>0.218</b>	<b>0.308</b>	<b>0.247</b>	0.576	0.094	0.162	0.100
	6	<b>0.711</b>	0.105	0.250	0.146	<b>0.686</b>	<b>0.096</b>	<b>0.241</b>	<b>0.135</b>
50000	2	0.711	0.105	0.250	0.146	0.661	0.132	<b>0.283</b>	0.170
	4	<b>0.719</b>	<b>0.132</b>	<b>0.262</b>	<b>0.170</b>	0.584	0.090	0.237	0.115
	6	0.711	0.105	0.250	0.146	<b>0.711</b>	<b>0.153</b>	<b>0.283</b>	<b>0.191</b>
100000	2	0.694	0.099	0.233	0.137	0.584	0.096	0.212	0.109
	4	<b>0.711</b>	<b>0.105</b>	<b>0.250</b>	<b>0.146</b>	<b>0.652</b>	<b>0.168</b>	<b>0.283</b>	<b>0.198</b>
	6	<b>0.711</b>	<b>0.105</b>	<b>0.250</b>	<b>0.146</b>	0.635	0.107	0.245	0.141

or not. In other words, the chosen motifs may not be the best representatives of class. This might be solved by finding the most frequent motifs as these might be a better indicator of class. Furthermore, it is conjectured that smoothing and filtering may contribute to classification effectiveness, as this will serve to reduce runtime and remove noise.

Given the results presented in the previous section, it can be concluded that the classification, using the MK algorithm and segmented and non-segmented time series, was similar (best accuracy values of approximately 70% were obtained). However, using PCGseg significantly less runtime was required; the runtime was improved by a factor of six. This runtime advantage is then the principal benefit offered by the proposed PCGseg approach.

## 7 Conclusions

In this paper, a novel time-series segmentation method, PCGseg, has been proposed for segmenting the PCG time series data for facilitating motif-based time-series analysis. More specifically, the MK algorithm can be used to identify appropriate motifs in the PCG data prior to the classification process. The objective of the segmentation was to reduce the amount of data, by removing fine details, so as to provide for a more tractable representation while retaining all salient features. The performance of the proposed approach was evaluated by applying it to a realistic multi-class PCG classification problem. The evaluation shows that the proposed approach has a very promising performance gains, as much as six times of speedup compared to the traditional approach, while offering an acceptable level of accuracy, within 70% of the best recorded accuracy. For future work, the authors would like to consider the usage of alternative motif-based time series classification techniques, more specifically techniques where motifs are selected according to frequency of occurrence. The intuition here is that such motifs will be better class differentiators in the context of PCG classification scenarios.

## References

1. Bezruchko, B., Smirnov, D.: *Extracting Knowledge From Time Series: An Introduction to Nonlinear Empirical Modeling*. Springer, Heidelberg (2010). <https://doi.org/10.1007/978-3-642-12601-7>
2. Chiu, B., Keogh, E., Lonardi, S.: Probabilistic discovery of time series motifs. In: *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 493–498. ACM (2003)
3. Gao, Y., Lin, J., Rangwala, H.: Iterative grammar-based framework for discovering variable-length time series motifs. In: *Proceedings of the 15th IEEE International Conference on Machine Learning and Applications (ICMLA 2017)*, pp. 111–116. IEEE (2017)
4. Serra, J., Arcos, J.: Cparticle swarm optimization for time series motif discovery. *Knowl.-Based Syst.* **92**, 127–137 (2016)
5. Torkamani, S., Lohweg, V.: Survey on time series motif discovery. *Wiley Interdiscip. Rev. Data Mining Knowl. Discov.* **7**(2), 1–8 (2017)
6. Keogh, E., Chu, S., Hart, D., Pazzani, M.: Segmenting time series: a survey and novel approach. In: Kandel, A., Bunke, H., Last, M. (eds.) *Data mining in Time Series Databases*, pp. 1–22. World Scientific (2001)
7. Huang, G., Zhou, X.: A piecewise linear representation method of hydrological time series based on curve feature. In: *Proceedings of the 8th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC 2016)*, pp. 203–207 (2016)
8. Anirudh, R., Turaga, P.: Geometry-based symbolic approximation for fast sequence matching on manifolds. *Int. J. Comput. Vis.* **116**(2), 161–173 (2016)
9. Keogh, E., Chakrabarti, K., Pazzani, M.: Mehrotra: dimensionality reduction for fast similarity search in large time series databases. *J. Knowl. Inf. Syst.* **3**(3), 263–286 (2001)
10. Patel, A., Bullmore, E.: A wavelet-based estimator of the degrees of freedom in denoised fmri time series for probabilistic testing of functional connectivity and brain graphs. *NeuroImage* **142**, 14–26 (2016)
11. Zhao, H., Dong, Z., Li, T., Wang, X., Pang, C.: Segmenting time series with connected lines under maximum error bound. *Inf. Sci.* **345**, 1–8 (2016)
12. Zhao, H., Li, G., Zhang, H., Xue, Y.: An improved algorithm for segmenting online time series with error bound guarantee. *Int. J. Mach. Learn. Cybern.* **7**(3), 365–374 (2016)
13. Belevich, I., Joensuu, M., Kumar, D., Vihinen, H., Jokitalo, E.: Microscopy image browser: a platform for segmentation and analysis of multidimensional datasets. *PLOS Biol. J.* **14**(1), 1–13 (2016)
14. Oliveira, J., Sousa, C., Coimbra, M.: Coupled hidden Markov model for automatic ECG and PCG segmentation. In: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2017)*, pp. 1023–1027 (2017)
15. Quiceno, A., Delgado, E., Vallverd, M., Matijasevic, A., Castellanos-Domnguez, G.: Effective phonocardiogram segmentation using nonlinear dynamic analysis and high-frequency decomposition. In: *Proceedings of the Computers in Cardiology*, pp. 161–164. IEEE (2008)
16. Ahlstrom, C.: *NonLinear Phonocardiographic Signal Processing*. Ph.D. thesis, Linkoping University, Sweden (2008)

17. Dokur, Z., Imez, T.: Heart sound classification using wavelet transform and incremental self-organizing map. *Digit. Sig. Process.* **18**(6), 951–959 (2008)
18. Gavrovska, A., Paskas, M., Dujkovic, D., Reljin, I.: Region-based phonocardiogram event segmentation in spectrogram image. In: *Proceedings of the Neural Network Applications in Electrical Engineering (NEUREL 2010)*, pp. 69–62. IEEE (2010)
19. Moukadem, A., Dieterlen, A., Hueber, N., Brandt, C.: Comparative study of heart sounds localization. In: *Proceedings of the Bioelectronics, Biomedical and Bio-inspired Systems, SPIE Proceedings*, vol. 8068, 9 p. (2011)
20. Helton, W.: *Canine Ergonomics: The Science of Working Dogs*. CRC Press, Boca Raton (2009)
21. Lin, J., Keogh, E., Wei, L., Lonardi, S.: Experiencing sax: a novel symbolic representation of time series. *Data Min. Knowl. Disc.* **15**(2), 107–144 (2007)
22. Sklansky, J., Gonzalez, V.: Fast polygonal approximation of digitized curves. *Pattern Recogn.* **12**(5), 327–331 (2007)
23. Mueen, A., Keogh, E., Zhu, Q., Cash, S., Westover, B.: Exact discovery of time series motifs. In: *Proceedings of the SIAM International Conference on Data Mining*, pp. 473–484 (2009)
24. Nakamura, K., Kawamoto, S., Osuga, T., Morita, T., Sasaki, N., Morishita, K., Takiguchi, M.: Left atrial strain at different stages of myxomatous mitral valve disease in dogs. *J. Vet. Intern. Med.* **31**(2), 316–325 (2017)
25. Chen, C., Pau, L., Wang, P.: *Handbook of Pattern Recognition and Computer Vision*. World Scientific, River Edge (1993)
26. Kuncheva, L.: *Combining Pattern Classifiers: Methods and Algorithms*, 2nd edn. Wiley, New York (2014)
27. Wang, X., Fang, Z., Wang, P., Zhu, R., Wang, W.: A distributed multi-level composite index for KNN processing on long time series. In: Candan, S., Chen, L., Pedersen, T.B., Chang, L., Hua, W. (eds.) *DASFAA 2017. LNCS*, vol. 10177, pp. 215–230. Springer, Cham (2017). [https://doi.org/10.1007/978-3-319-55753-3\\_14](https://doi.org/10.1007/978-3-319-55753-3_14)
28. Stojanovic, M., Bozic, M., Stankovic, M., Stajic, Z.: A methodology for training set instance selection using mutual information in time series prediction. *Neurocomputing* **141**, 236–245 (2014)
29. Witten, I., Frank, E., Hall, M., Pal, C.: *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann (2016)



# A Lazy One-Dependence Classification Algorithm Based on Selective Patterns

Zhuoya Ju<sup>1</sup>, Zhihai Wang<sup>1</sup>(✉), and Shiqiang Wang<sup>2</sup>

<sup>1</sup> Beijing Jiaotong University, Beijing 100044, China  
{juzhuoya,zhhwang}@bjtu.edu.cn

<sup>2</sup> 1Verge Internet Technology (Beijing) Co., Ltd., Beijing, China  
wangshiqiang@alibaba-inc.com

**Abstract.** Data mining is a widely acceptable method on mining knowledge from large databases, and classification is an important technique in this research field. A naïve Bayesian classifier is a simple but effective probabilistic classifier, which has been widely used in classification. It is commonly thought to assume that the probability of each attribute belonging to a given class value is independent of all other attributes in the naïve Bayesian classifier; however, there are lots of contexts where the dependencies between attributes are complex and should thus be considered carefully. It is an important technique that constructing a classifier using specific patterns based on “attribute-value” pairs in lots of researchers’ work, and the classification result will be impacted by dependencies between these specific patterns meanwhile. In this paper, a lazy one-dependence classification algorithm based on selective patterns is proposed, which utilizes both the patterns’ discrimination and dependencies between attributes. The classification accuracy benefits from mining and employing patterns which own high discrimination, and building the one-dependence relationship between attributes in a proper way. Through an exhaustive experimental evaluation, it shows that the proposed algorithm is competitive in accuracy with the state-of-the-art classification techniques on datasets from the UCI repository.

**Keywords:** Classification · Pattern discovery · Dependence Bayesian classifier · Lazy learning

## 1 Introduction

In the machine learning field and data mining technology, classification is regarded as a crucial learning method. Classification algorithms based on Bayesian network own a solid theoretical basis, strong anti-noise performance, good classification performance and robustness. A naïve Bayesian classifier is commonly thought to assume that the probability of each attribute belonging

---

Supported by Beijing Natural Science Foundation (4182052), and National Natural Science Foundation of China (61672086, 61771058).

to a given class value is independent of all other attributes, and this assumption is so called “conditional independence assumption” [1]. However, there are lots of contexts where the dependencies between attributes are complex and should thus be considered carefully. To accord with actual situation, dependencies between attributes are researched deeply in many classification algorithms: Tree Augmented Naïve Bayes Classifier (TAN) [2], Aggregating One-Dependence Estimators (AODE) [3], and its extended algorithms [4–6], and so on. Nevertheless, it does not delve into that the key role of patterns based on attributes and their values in these classification algorithms.

In general, an instance is characterized by  $n$  ( $n=1, 2, 3, \dots$ ) pairs of “attribute-value” which are called “items”. Each instance without missing values can be considered as an itemset with  $n$  items. There has been a great deal of research on the use of itemsets to complete the classification tasks and achieved high classification accuracy already. For instance, Classification by Aggregating Emerging Pattern (CAEP) [7], the Bayesian Classification based on Emerging Patterns (BCEP) [8], classifiers based on Jumping Emerging Patterns (JEPs) [9], and so on.

In order to exploit the patterns’ discrimination and construct dependencies between attributes using Bayesian networks, this paper proposes a lazy one-dependence classification algorithm based on the selective patterns. The selective patterns (such as frequent patterns, emerging patterns, etc.) that play major roles as the basis for classification are mined first, and then two types of attributes (belonging to the selective patterns or not) are analyzed by the Bayesian network which constructs a one-dependence relationship. After that a new classification model is built.

## 2 Background

The classification based on the selective patterns is to find out some specific patterns with high growth rates from non-target to target class, and analyze dependencies between the attributes contained in these specific patterns and other attributes. Some rudimentary definitions and formulas are given below.

The *support* of itemset  $I$ ,  $Supp_D(I) = count_D(I)/|D|$ , where  $count_D(I)$  is the number of instances in dataset  $D$  that contains itemset  $I$ , and  $|D|$  is the total number of instances in dataset  $D$ .

Given a dataset  $D$ , which is divided into two different subsets  $D_1$  and  $D_2$ , namely,  $D_1 \cap D_2 = \emptyset$ ,  $D_1 \cup D_2 = D$ . The *growth rate* of itemset  $I$  from  $D_1$  to  $D_2$ , namely  $Growth(I, D_1, D_2)$ , is defined as follows,

$$Growth(I, D_1, D_2) = \begin{cases} 0 & Supp_{D_1}(I) = Supp_{D_2}(I) = 0 \\ \infty & Supp_{D_1}(I) = 0, Supp_{D_2}(I) > 0 \\ \frac{Supp_{D_2}(I)}{Supp_{D_1}(I)} & others \end{cases} \quad (1)$$

A *Selective pattern* is the itemset  $I$  whose support  $Supp_{D_1}(I)$  and growth rate  $Growth(I, D_1, D_2)$  satisfy the threshold  $\xi, \rho$  respectively, and thus it owns discrimination to classify instances.

Patterns represent the nature and important characteristics of datasets and form the basis of many important data mining tasks. The boundary algorithm BORDER\_DIFF proposed by Dong et al. [10] is used to mine specific patterns for classification in this paper. Removal of redundant patterns and noises will help to speed up the classification and improve the classification accuracy. In this paper, the method in [8] is used to filtering patterns.

The aggregation one-dependence estimator (AODE) averages all models from a restricted class of one-dependence classifiers, the class of all such classifiers that have all other attributes depend on a common attribute and the class [3]. Each instance can be described by an  $n$ -dimensional attribute vector  $X = (a_1, a_2, \dots, a_n)$ , where  $a_i$  represents the value of the  $i$ th attribute  $A_i$ . Given instance  $X$ , the task of classification is to calculate the class of the maximum a posteriori (MAP) as  $X$ 's prediction class, which can thus be expressed as:

$$P(c_k|X) \propto \frac{\sum_{i:1 \leq i \leq n \wedge F(a_i) \geq u} P(c_k, a_i) \cdot \prod_{j=1, j \neq i}^n P(a_j|c_k, a_i)}{|i : 1 \leq i \leq n \wedge F(a_i) \geq u|} \quad (2)$$

where  $F(a_i)$  is a count of the number of training examples having attribute-value  $a_i$  and is used to enforce the limit  $u$  placed on the support needed in order to accept a conditional probability estimate.

### 3 A Lazy Classification Algorithm Based on Selective Patterns

According to Bayes theorem,  $P(c|X)$  can be expressed as:

$$P(c|a_1, a_2, \dots, a_n) = \frac{P(a_1, a_2, \dots, a_n|c)P(c)}{P(a_1, a_2, \dots, a_n)} \propto P(a_1, a_2, \dots, a_n|c)P(c) \quad (3)$$

Given a class  $c$ , assuming that the attributes contained in the pattern and other attributes are conditionally independent of each other, namely:

$$\begin{aligned} P(c|a_1, a_2, \dots, a_n) &\propto P(c)P(a_1, a_2, \dots, a_i|c)P(a_{i+1}, \dots, a_n|c) \\ &= P(c|a_1, a_2, \dots, a_i)P(a_1, a_2, \dots, a_i)P(a_{i+1}, \dots, a_n|c) \end{aligned} \quad (4)$$

where  $\{a_1, a_2, \dots, a_i\}$  represents the attributes included in the pattern.

#### 3.1 Characterization of Discriminative Patterns

Assume that the set of attributes contained in itemset  $e$  is  $\{a_1, a_2, \dots, a_i\}$ , which is a special pattern of a growth rate of  $Growth(e, c', c)$  from a data set of class  $c'$  to a data set of class  $c$ . When a test instance contains  $e$ , the probability that this instance belongs to class  $c$  is  $P(c|a_1, a_2, \dots, a_i)$ , and abbreviated as  $P(c|e)$ . According to the definitions of *support* and *growth rate*, then:

$$P(c|e) = \frac{Growth(e, c', c)}{Growth(e, c', c) \frac{|c|}{|c'|} + 1} \frac{|c| + |c'|}{|c'|} P(c) \quad (5)$$

### 3.2 A Lazy One-Dependence Classification Algorithm

The Aggregate One-Dependence Classification based on Selective Patterns (AODSP) is proposed in this paper. Attributes in a specific pattern are treated as a whole, and the attributes in the pattern are assumed to be independent of attributes out of the pattern. AODSP assumes that the dependencies between attributes out of a specific pattern satisfy a one-level Bayesian tree structure, that is, each attribute sequentially serves as the parent of other attributes and the remaining attributes depend on this attribute as child nodes. The average probability calculated by the multiple classifiers is as the classification probability.

Let  $E$  be a set of all patterns, and the attributes in a pattern  $e$  which is contained in  $E$  are treated as a whole. From the Eq. 2, the conditional probability of attributes out of  $e$  satisfies:

$$P(a_{i+1}, \dots, a_n | c) = \frac{P(c | a_{i+1}, \dots, a_n) P(a_{i+1}, \dots, a_n)}{P(c)} \propto \frac{P(c | a_{i+1}, \dots, a_n)}{P(c)} \quad (6)$$

$$\propto \frac{\sum_{j: i+1 \leq j \leq n \wedge F(a_j) \geq u} P(c, a_j) \prod_{k=i+1, k \neq j}^n P(a_k | c, a_j)}{|j : i+1 \leq j \leq n \wedge F(a_j) \geq u|} \frac{1}{P(c)}$$

and then:

$$P(c | a_1, a_2, \dots, a_n) \propto P(c | a_1, a_2, \dots, a_i) P(a_1, a_2, \dots, a_i) P(a_{i+1}, \dots, a_n | c)$$

$$\propto P(c | a_1, a_2, \dots, a_i) P(a_1, a_2, \dots, a_i) \quad (7)$$

$$\cdot \frac{\sum_{j: i+1 \leq j \leq n \wedge F(a_j) \geq u} P(c, a_j) \prod_{k=i+1, k \neq j}^n P(a_k | c, a_j)}{|j : i+1 \leq j \leq n \wedge F(a_j) \geq u|} \frac{1}{P(c)}$$

The patterns' discrimination is applied to the Bayesian network, that is, the Eq. 5 is substituted into the Eq. 7 and the discrimination of all patterns is aggregated to obtain the probability prediction equation adopted by the aggregation one-dependent classification algorithm based on selective patterns:

$$P(c | a_1, a_2, \dots, a_n) \propto \sum_{e \in E} \left( \frac{Growth(e, c', c)}{Growth(e, c', c) \frac{|c|}{|c'|} + 1} \frac{|c| + |c'|}{|c'|} \cdot P(a_1, a_2, \dots, a_i) \right.$$

$$\left. \cdot \frac{\sum_{j: i+1 \leq j \leq n \wedge F(a_j) \geq u} P(c, a_j) \prod_{k=i+1, k \neq j}^n P(a_k | c, a_j)}{|j : i+1 \leq j \leq n \wedge F(a_j) \geq u|} \right) \quad (8)$$

Equation 8 aggregates the discrimination of all patterns, and further considers the dependencies between attributes out of patterns. When classifying a test instance, the class of the instance will be the class that maximizes the Eq. 8.

The AODSP algorithm is defined in Algorithm 1.

## 4 Experiments and Evaluations

In order to validate the accuracy of the proposed aggregation one-dependent classification algorithm based on the selective patterns, 8 datasets from the UCI repository of machine learning databases [11] are used as experimental datasets (see Table 1).



**Algorithm 1.** AODSP(*instance*, *m*, *E*)**Input:***instance*: to be classified; *m*: the number of class labels; *E*: sets of patterns.**Output:***probs*: distribution of prediction class labels of *instance*.

```

1: for  $i \in [0, m]$  do
2:    $probs[i] = 0;$ 
3:   for  $j \in [0, E[i].size())$  do
4:     Itemset  $e = (Itemset)E[i].elementAt(j);$ 
5:     if instance contains  $e$  then
6:       calculate  $P(i|attributes\ in\ e);$ 
7:       calculate  $P_{AODE}(attributes\ not\ in\ e|i);$ 
8:       calculate  $P(a_1, a_2, \dots, a_i)$ , where  $a_1, a_2, \dots, a_i \in e;$ 
9:        $p = P(i|instance);$ 
10:    end if
11:     $probs[i] = probs[i] + p;$ 
12:  end for
13:  if  $probs[i] \leq 0$  then
14:     $probs[i] = aode.distributionForInstance(instance);$ 
15:  end if
16: end for
17: Normalize probs;
18: return probs;

```

#### 4.1 Parameter Analysis

Patterns' discrimination is determined by the growth rate and the supports in different classes. Table 2 shows the error rates of AODSP on the iris dataset with different supports. Wherein, the value of "S" means the random seed used in the experiment; "Mean" means the average value of 5 results; "Support1" and "Support2" respectively represent the support of patterns on the non-target class and the target class; "Growth Rate" means the growth rate from non-target class to target class.

It can be figured out that: when the support is set too high, selective patterns can't be found, AODSP degenerates to AODE; when the support is set too low, there will be too many patterns which may lead to overfitting and the error rate is compromised.

#### 4.2 Empirical Setup

For numerical attributes, the multi-interval discretization method provided in [12] is adopted to discrete them as a preprocessing step. The experiment uses a 10-fold cross-validation method to calculate the error rate of the classifiers. The random seed is set as 1, 2, 3, 5, and 7 respectively to calculate the error rate, and the average is taken as classification results. The AODSP is compared with the NB, AODE, ASAODE, and BCEP in error rate. AODSP takes the

**Table 1.** Summary of datasets.

No	Domain	Data file	Instances	Attributes	Classes	Missing value
1	Balance Scale	balance-scale	625	4	3	No
2	Liver Disorders	bupa	345	6	2	No
3	German	german	1000	20	2	No
4	House Votes 84	house-votes-84	435	16	2	No
5	Iris Classification	iris	150	4	3	No
6	Labor Negotiations	labor	57	16	2	Yes
7	New-Thyroid	new-thyroid	215	5	3	No
8	Pima Indians Diabetes	pid	768	8	2	No

Note: The number of “Attributes” does not include the class attribute.

**Table 2.** Error rates of AODSP on iris with respect to different supports.

No	$S = 1$	$S = 2$	$S = 3$	$S = 5$	$S = 7$	Mean	Support1	Support2	Growth rate
1	6.67	6.67	7.33	6.00	6.00	6.53	0.05	0.8	16
2	5.33	6.00	5.33	4.00	6.00	5.33	0.05	0.5	10
3	4.67	4.67	4.67	4.67	4.67	4.67	0.1	0.5	5
4	5.33	5.33	5.33	6.00	5.33	5.47	0.2	0.4	2
5	6.67	6.67	7.33	7.33	6.00	6.80	0.4	0.6	1.5
6	6.67	6.67	7.33	7.33	6.00	6.80	0.5	0.6	1.2

support and growth threshold the same as BCEP (namely, the minimum support threshold  $\xi = 1\%$  or an absolute count of 5; the minimum growth rate  $\rho = 5$ ).

### 4.3 Error Rate Analysis

Table 3 shows the error rates of the 5 classifiers on 8 data sets, with the lowest error rate in bold. The AODSP based on selective patterns proposed in this paper adopts the idea of AODE to deal with the dependencies between attributes in and out of specific patterns. ASAOE is an improvement of AODE, and BCEP tries to combine emerging patterns with Bayesian network. They all belong to Bayesian network and are picked as reference objects.

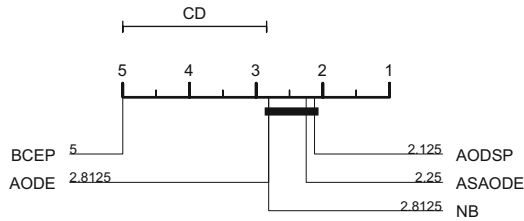
According to the experiment, the following conclusions can be figured out: Compared to AODE, AODSP achieves lower error rate on 5 datasets and the same error rate on 1 datasets; Compared to BCEP, AODSP achieves lower error rate on all 8 datasets; Compared to other classifiers, AODSP achieves the lowest average error rate on all 8 datasets.

The experiment uses the method proposed by Demšar [13] to test the critical difference of the classifiers on 8 datasets in Fig. 1, with a significance level of 0.05. It can be figured out that: The accuracy of AODSP is significantly higher, and the classification accuracy is greatly improved comparing to AODE and BCEP.

**Table 3.** Comparison among algorithms' error rate (%).

Data file	NB	AODE	ASAODE	BCEP	AODSP
balance-scale	<b>28.80</b>	30.40	29.44	54.24	30.11
bupa	36.81	36.81	36.81	42.03	36.81
german	24.66	<b>23.44</b>	23.80	25.50	24.22
house-votes-84	9.93	5.79	5.52	10.24	<b>5.20</b>
iris	5.60	6.80	6.00	33.33	<b>4.67</b>
labor	7.72	7.37	<b>5.26</b>	10.00	<b>5.26</b>
new-thyroid	<b>3.72</b>	4.19	5.58	30.23	4.47
pid	22.16	21.98	<b>21.22</b>	23.20	21.90
average	17.43	17.10	16.70	28.60	<b>16.58</b>

All classifiers except BCEP are part of a single clique, namely, most classifiers do not obtain significantly higher or lower results in terms of accuracy.

**Fig. 1.** Critical difference diagram for different classifiers on the 8 data sets.

## 5 Conclusion and Future Work

The naïve Bayesian classification is effective but restrictive due to its conditional independence assumptions. In order to weaken the conditional independence of the naïve Bayesian classification algorithm, in this paper, attributes selection is made based on the patterns composed of “attribute-value” pairs, and a lazy one-dependence classification algorithm based on selective patterns is proposed. The discrimination of selective patterns that play major roles as the basis for classification is mined, and the relationship between two types of attributes (belonging to selective patterns or not) is analyzed by the Bayesian network. The accuracy of the proposed algorithm is verified by using 8 datasets in the UCI machine learning database as experimental data. Based on the further analysis of the experimental results, it proves that the classification ability can be effectively improved by using the discrimination of patterns and dealing with dependencies between the attributes. In future work, it is necessary to further consider dependencies between the attributes and explore more appropriate patterns' selection criteria.

## References

1. Domingos, P., Pazzani, M.: Beyond independence: conditions for the optimality of the simple Bayesian classifier. In: Saitta, L. (ed.) *Proceedings of the 13th ICML*, pp. 105–112. Morgan Kaufmann, San Francisco (1996)
2. Friedman, N., Goldszmidt, M.: Building classifiers using Bayesian networks. In: *Proceedings of the 13th National Conference on Artificial Intelligence (AAAI 1996)*, pp. 1277–1284. AAAI Press, Menlo Park (1996)
3. Webb, G., Boughton, J., Wang, Z.: Not so naïve Bayes: aggregating one-dependence estimators. *Mach. Learn.* **58**(1), 5–24 (2005)
4. Chen, S., Martínez, A.M., Webb, G.I.: Highly scalable attribute selection for averaged one-dependence estimators. In: Tseng, V.S., Ho, T.B., Zhou, Z.-H., Chen, A.L.P., Kao, H.-Y. (eds.) *PAKDD 2014. LNCS (LNAI)*, vol. 8444, pp. 86–97. Springer, Cham (2014). [https://doi.org/10.1007/978-3-319-06605-9\\_8](https://doi.org/10.1007/978-3-319-06605-9_8)
5. Chen, S., Martínez, A.M., Webb, G.I., Wang, L.: Selective AnDE for large data learning: a low-bias memory constrained approach. *Knowl. Inf. Syst.* **50**(2), 475–503 (2017)
6. Yu, L., Jiang, L., Wang, D., Zhang, L.: Attribute value weighted average of one-dependence estimators. *Entropy* **19**(9), 501 (2017)
7. Dong, G., Zhang, X., Wong, L., Li, J.: CAEP: classification by aggregating emerging patterns. In: Arikawa, S., Furukawa, K. (eds.) *DS 1999. LNCS (LNAI)*, vol. 1721, pp. 30–42. Springer, Heidelberg (1999). [https://doi.org/10.1007/3-540-46846-3\\_4](https://doi.org/10.1007/3-540-46846-3_4)
8. Fan, H., Ramamohanarao, K.: A Bayesian approach to use emerging patterns for classification. In: Schewe, K., Zhou, X. (eds.) *Proceedings of the 14th Australasian Database Conference*, pp. 39–48. ACS Press, Adelaide, Australia (2003)
9. Li, J., Dong, G., Ramamohanarao, K.: Making use of the most expressive jumping emerging patterns for classification. *Knowl. Inf. Syst.* **3**(2), 131–145 (2001)
10. Dong, G., Li, J.: Efficient mining of emerging patterns: discovering trends and differences. In: *Proceedings of the 5th ACM SIGKDD International Conference on KDD*, pp. 43–52. ACM Press, New York (1999)
11. Blake, C., Merz, C.: UCI repository of machine learning databases. <http://archive.ics.uci.edu/ml/index.html>. Accessed 1 June 2018
12. Fayyad, U.M., Irani, K.B.: Multi-interval discretization of continuous valued attributes for classification learning. In: Bajcsy, R. (ed.) *Proceedings of the 13th International Joint Conference on Artificial Intelligence*, pp. 1022–1027. Morgan Kaufmann, San Mateo, CA (1993)
13. Demšar, J.: Statistical comparisons of classifiers over multiple data sets. *J. Mach. Learn. Res.* **7**(1), 1–30 (2006)



# A Client-Assisted Approach Based on User Collaboration for Indoor Positioning

Yiyi Zhang<sup>1</sup>, Jun Tang<sup>2</sup>, Michael Elimu<sup>1</sup>, and Naizheng Bian<sup>1</sup>(✉)

<sup>1</sup> College of Computer Science and Electronic Engineering,  
Hunan University, Changsha 410082, China  
18390980140@139.com

<sup>2</sup> College of Electrical and Information Engineering,  
Hunan University, Changsha 410082, China

**Abstract.** Fingerprint indoor positioning technology is one of the most attractive and promising techniques for mobile devices positioning. However, it is also time consuming for building the radio map offline and cannot provide reliable accuracy due to the changing environment. In response to this compelling problem, a client-assisted (CA) approach is proposed for radio map construction based on multi-user collaboration. In this method, multi-dimensional scaling (MDS) approach is used to transform the distance to two dimensional data. MDS as an set of analytical technique has been used for many years in fields like economics and marketing research. It is a suitable for reducing the data dimensionality to points in two or three dimensional space. In CA, this can be used where only distances between users are known which are used as an input data. All the client data is collected at one point because of the centralized nature of the MDS. It is advantageous in that, MDS can reconstruct the relative map of the network even when there are no anchor clients (clients with a priori known location). Given a sufficient number of known client locations, MDS generates accurate position estimation enabling local map to be transformed into an absolute map.

Based on gradient features of users' walking speed, solve-stuck (SS) method is adopted to improve the efficiency by reducing calculation complexity and solving "data drift" problem. Radio map with a small number of labeled fingerprints can be self-updated by iterating the distance between users. Kalman filter (KF) method is used to remove the noise to make the trajectory closer to the ideal trajectory. We further demonstrate the influence of density distribution and time-cost of different number of clients. The experimental results show that CA approach can improve positioning accuracy with acceptable time-cost.

**Keywords:** Indoor localization · Wi-Fi fingerprint · Client-assisted  
Radio map

## 1 Introduction

With the improvement of communication technology and increase of user demands, precise location services are playing an important role in service market, such as tourism. Currently, outdoor positioning technology (OPT) is fully developed, while there is still

much room to improve indoor positioning technology (IPT). For OPT, the use of GPS can meet location accuracy of meters in both online and offline mode, but it is weak in indoor due to obstacles and multi-path effect. Therefore, the research of IPT focuses on improving the limited coverage using APs instead of expensive devices and time consumption of setting up the radio map.

The existing IPT mainly uses short-range wireless signals, such as Bluetooth [1], RFID [2], ultra-wideband signals [3], etc. Although these techniques can achieve high accuracy, the implementation complexity and cost are great, which is not suitable for large-scale application [4]. With emergence of Wi-Fi network access points (APs) in public places, Wi-Fi based received signal strength indicator (RSSI) positioning technology ease positioning pressure by saving the cost of positioning equipment deployment [5]. A Wi-Fi fingerprint-based indoor localization technique was proposed to deal with complicated problems without additional equipment [6]. A method [7] was proposed to construct a graph-based semi-supervised radio map with the substantially reduced fingerprints calibration cost in offline phase. In [8], a graph-based semi-supervised learning (G-SSL) model based on 1-graph algorithm were presented to reduce the labor and time consumption. An unsupervised method was applied to reduce such manual efforts by exploiting unlabeled fingerprints [9]. In [10], the crowdsensing data based on large number of APs was applied to reduce such manual efforts by holistically treating the data. However, the user collaboration and distances between APs are not fully evaluated in this research area. Furthermore, the radio map constructed cannot self-update over time, and thus reduces the robustness of the fingerprint.

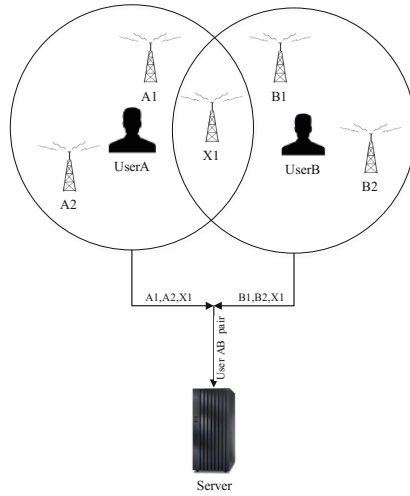
In this paper, a CA approach based on user collaboration is proposed to improve the positioning accuracy of radio map by iterating the distance between users. It can make full use of huge user resources and generate radio maps that can be automatically updated. It can reduce the labour and time cost by using the small labeled data of radio map. SS method is used to simplify complexity and solve the problem of “data drifting”. The CA system doesn’t require any additional device but rather, a smart phone and an indoor floor plan making its implementation easy in many indoor scenarios.

## 2 Proposed Approach

### 2.1 CA System Model

CA approach has three basic steps: scanning, searching and correcting step. In the scanning step, location server collects users’ RSS information and analyzes to find out which users can be paired. In the searching step, Dijkstra’s algorithm is used to find the mini-hops AP chains between the user pair and calculate the distances. In the correcting step, the distance between users is transformed to the coordinate in fingerprint by MDS Algorithm [11], which is used to update radio map and correct positioning results. MDS consists of 3 (1) Calculate shortest distances between every pair of nodes (using either Dijkstra’s or Floyd’s all pairs shortest path algorithm). This is the distance matrix

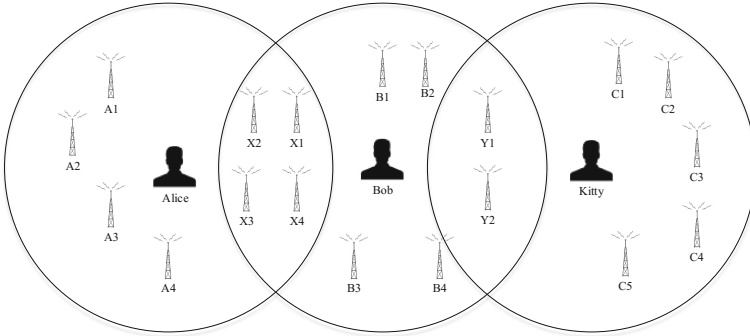
that serves as input to the multidimensional scaling in 2. (2) Apply classical multidimensional scaling to the distance matrix. The first two largest eigenvalues and eigenvectors give a relative map with relative location for each client. (3) Transform the relative map into absolute map using sufficient number of anchor clients (at least 3). The process is shown in Fig. 1.



**Fig. 1.** Overview of the proposed system.

where user A is an object who needs positioning service and user B is an assisting positioning object. If both user A and user B are within the coverage of at least one AP, they are defined as user AB pair by the positioning server. Reference point (RP) can be seen as virtual user. For an AP chain, {UserA, A1, B1, UserB}, the distance between A1 and B1 can be calculated by AP coordinate. The hybrid ToA/AoA approach [12] is adopted to calculate the distance between user and AP.

The number and location of the APs are assumed to be unchanged when fingerprint pattern are constructed, and thus the distance between APs are certain, as shown in Fig. 2.



**Fig. 2.** The overlap of transmission range.

For Alice, Bob and Kitty, the APs that can be detected are  $\{A1-4, X1-4\}$ ,  $\{X1-4, B1-4, Y1-2\}$  and  $\{Y1-2, C1-5\}$ , respectively. There are many mini-hops AP chains between each two users and the distance calculated by these AP chains will also be a lot. For example, the distance between Alice-Kitty can be estimated as

$$D_{\text{Alice-Kitty}} = D_{\text{Alice-X}} + D_{\text{X-Y}} + D_{\text{Y-Kitty}} \tag{1}$$

The linear regression method is used to remove the abnormal values when AP positions change. For example, the distance between Alice and Bob can be estimated as:

$$D_{\text{Alice-Bob}} = D_{\text{Alice-X}(1-4)} + D_{\text{X}(1-4)-B(1-4)} + D_{\text{B}(1-4)-\text{Bob}} \tag{2}$$

The MDS algorithm which needs at least three RPs to locate  $(x, y)$  is adopted to transform the distance to two-dimensional plane. It can be used to modify the positioning result and the radio map can be updated over time. This method can maintain a high positioning accuracy despite a small number of AP changes.

## 2.2 CA Algorithm

In multi-user cases,  $K$  users are selected as corporation nodes to join in the positioning process. The best  $K$  value is discussed in Sect. 3. The CA algorithm is shown below.



---

**Algorithm :**

---

**Mobile Device (Client):**

```

{
Scan the AP which a user can connect to in this
area
while(the User stops moving){
Send the User AB pair (UA , UB ) to positioning
Server;
Receive the msg with distance d(UA , UB );
}
Send the User AB pair (UA , UB ) to positioning
Server;
}

```

**Server:**

```

{
for K nodes
for AP i in UA 's latest scan results
for AP j in UB 's latest scan results
{
find path with mini-hops by Dijkstra's algorithm;
d(UA,UB)=d(UA, APi)+d (AP i , AP j ) +d(APj,
UB);
list.append(d(UA,UB));
}
for (int count:list){
remove the noise by linear regression method;
d=d+d(UA,UB);
}
d(UA,UB)=d/list.size();
find the location by RSSI in fingerprint
User(x,y)R by knn
transform d(UA,UB) to coordinates
User(x,y)F in fingerprint=MDS(d(UA,UB )) ;
User(x,y)= Mean(User(x,y)F, User(x,y)R);
)
Update.radio map
Send to client(User(x,y));
}

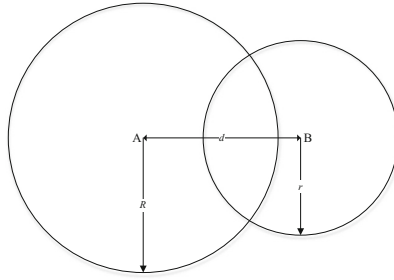
```

---

### 3 Influence of Density Distribution

#### 3.1 Effects of AP Density Distribution

In the realistic scenario, the distribution density of users and APs have influences on the positioning accuracy. When multiple users are within range of two APs with different propagation radius, as shown in the Fig. 3.



**Fig. 3.** The overlap 1 of AP A and B.

where  $A$ ,  $B$  are two APs, and  $R$ ,  $r$  are the coverage radius of them, respectively. Distance between  $A$  and  $B$  can be obtained as follow:

$$D_{AB} = f(R, r, M(A), M(B), M(A, B)) \quad (3)$$

where  $M(A)$  and  $M(B)$  are the number of users within the range of  $A$  and  $B$ , respectively.  $M(A, B)$  is the number of users within the overlapped range of  $A$  and  $B$ .

#### 3.2 The Influence of User Distribution

User distribution also has impacts on the positioning accuracy and the effect of random distribution and uniform distribution are researched.

The closed square area for the experiment is set to 20 m \* 15 m for different numbers of users and there are 3 APs in the area. The experiment result is shown in Fig. 4.

As the number of cooperation users increases, the positioning accuracy improves. When the number of users participating in co-location increases, the number of iterations increases, and the positioning accuracy increases with linear regression method. The accuracy of uniform distribution is better than random distribution. However, in a real scenario, users are free to move around, and the random distribution is close to the real distribution.

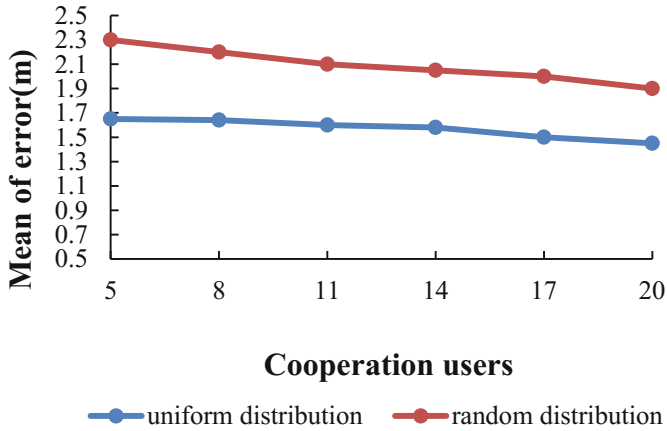


Fig. 4. The comparison of uniform distribution and random distribution.

## 4 Implementation and Evaluation

### 4.1 Environmental Setup

A set of experiments are conducted for CA approach mainly on the positioning accuracy, robustness, as well as the time cost and compared with the existing ones using the fingerprints solely. The room size is set to 20 m \* 15 m and the 3 APs are uniformly distributed in the room. The Ray-tracing method [13] is used to track the RSS for a location and spacing of the position points is set to 0.01 m. The comparison of positioning results for the different methods are shown in Fig. 5.

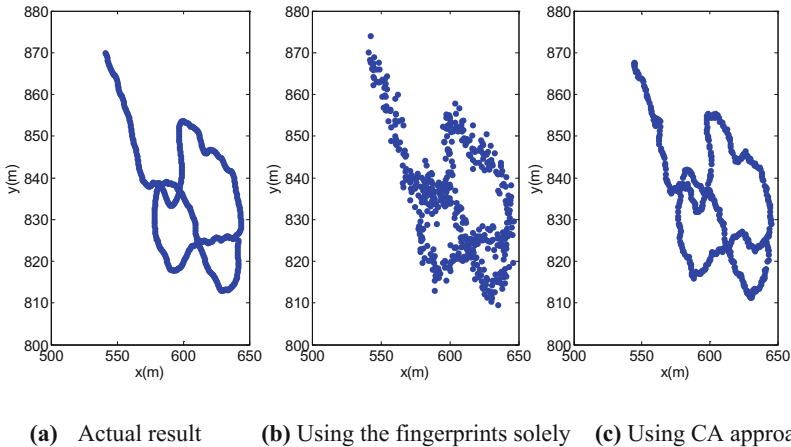
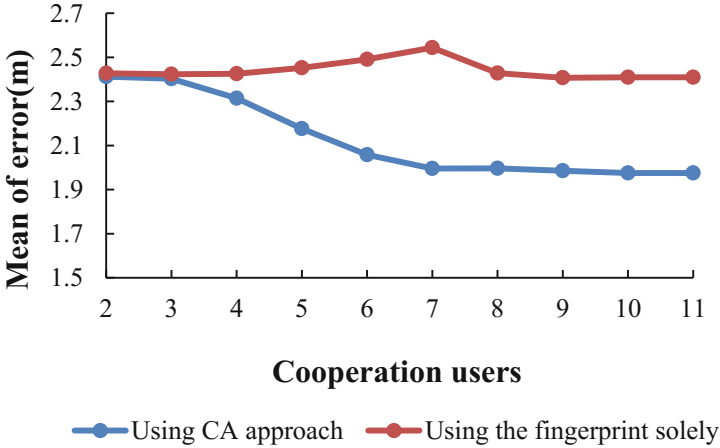


Fig. 5. Comparison of positioning results using different methods.

Compared with curve (b), curve (c) gets closer to the actual result in (a). After using the CA approach, the positioning accuracy is obviously better than using fingerprint solely.

The number of cooperating users also affect the positioning accuracy, as shown in Fig. 6.



**Fig. 6.** Comparison of error in different methods.

As the number of cooperation users increases, the error decreases. When the number is over 7, the positioning accuracy seems to be unchanged.

## 4.2 Efficiency Improvement

The CA approach needs to calculate the distance between users in real-time. The more the users participants, the greater the amount of iteration process needed. When the number of users reaches 20, stuck phenomenon such as “data-drifting”, “wall crash” occur. SS method is used to solve the problem and it can detect the users’ movement state and decide whether to use CA approach, the improvement effect is shown in Fig. 7. In addition, the positioning data will drift along with the walking distance which results in noise. KF method is adopted to remove noise, which makes the user trajectory smooth.

Compared with using fingerprint solely, the combination of SS and KF method makes the positioning closer to the actual result.

The more cooperation users participate in the process, the more iterations and time cost increases, as shown in Fig. 8.

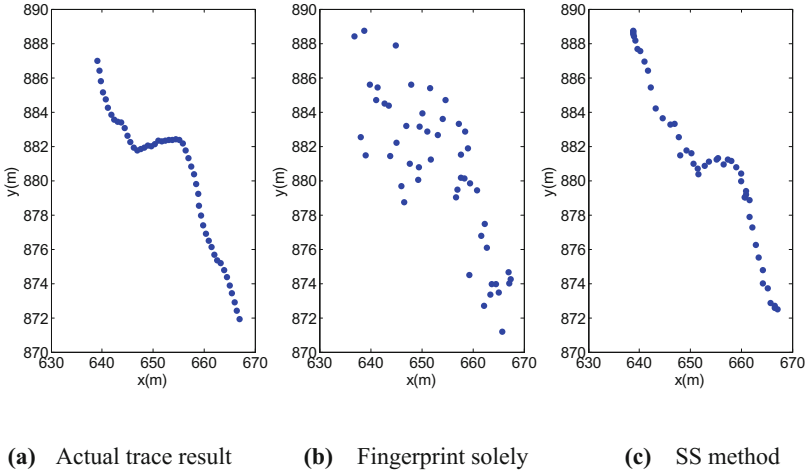


Fig. 7. Comparison of trajectory in two methods.

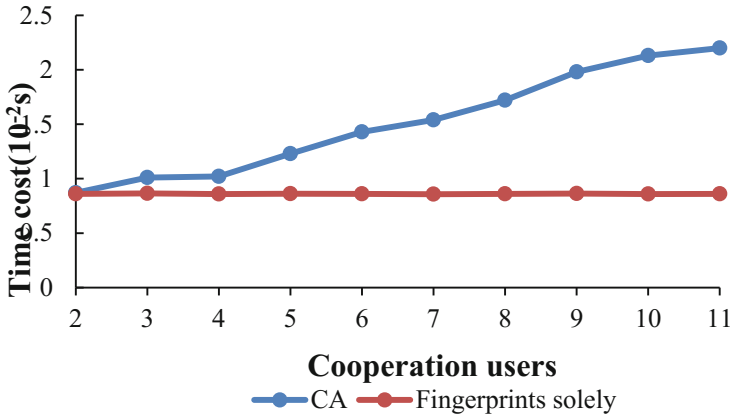


Fig. 8. Comparison of time cost in different methods.

The efficiency of positioning is formulated as follows:

$$\text{efficiency} = \frac{\text{accuracy}}{\text{time\_cost}} * 100\% \tag{4}$$

when the number of K nodes coordinated by users is 7, the efficiency reaches the best value via calculation. The average positioning accuracy is about 1.98 m and time cost is between 1.6–2.2 s.

The comparison of the positioning accuracy in different methods is shown in Table 1.

**Table 1.** Comparison of the positioning accuracy in different methods.

Algorithm	Accuracy	Algorithm	Accuracy
KNN	2.24 m	Gradient boosting for regression	2.22 m
Logistic regression	3.09 m	Multi-layer perceptron regressor	2.45 m
Support vector machine	2.25 m	CA approach	1.98 m
Random forest	2.21 m		

Compared with other methods, the accuracy of CA approach is higher.

## 5 Conclusion and Future Work

In this paper, we propose CA approach, a new solution for improving the positioning accuracy of radio map based on user collaboration. The fixed APs location is applied to calculate the distance between users, and then MDS method is used to transform the distance to coordinate, which corrects the original result. The comparative experiment shows that CA approach outperforms other methods, and the average accuracy is increased by 11.6% compared with the method of using fingerprint solely.

Future work includes extensive testing and investigation of CA approach in a more complicated indoor environment, such as multi-floor environment.

## References

1. Roy, S., Sarkar, S., Tah, A.: A Bluetooth-based autonomous mining system. In: Mohapatra, D.P., Patnaik, S. (eds.) *Intelligent Computing, Networking, and Informatics*. AISC, vol. 243, pp. 57–65. Springer, New Delhi (2014). [https://doi.org/10.1007/978-81-322-1665-0\\_6](https://doi.org/10.1007/978-81-322-1665-0_6)
2. Bahld, P., Padmanabhan, V.N.: RADAR: an in-building RF-based user location and tracking system. In: *Proceedings of the IEEE Conference on Computer and Communications Societies*, vol. 2, pp. 775–784 (2000)
3. Eleni, B., Demosthenes, V., Nikalaos, N.: Localization error modeling of hybrid fingerprint-based techniques for indoor ultra-wide band systems. *Telecommun. Syst.* **63**(2), 223–241 (2016)
4. Sorour, S., Lostanlen, Y., Valaee, S., Majeed, K.: Joint indoor localization and radio map construction with limited deployment load. *IEEE Trans. Mob. Comput.* **14**(5), 1031–1043 (2015)
5. Tran, D.A., Zhang, T.: Fingerprint-based location tracking with hodrick-prescott filtering. In: *Wireless and Mobile Networking Conference (WMNC)*, pp. 1–8 (2014)
6. Le, T.D., Le, H.M., Nguyen, N.Q., Tran, D., Nguyen, N.T.: Convert Wi-Fi signals for fingerprint localization algorithm. In: *2011 7th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM)*, pp. 1–5, 23–25 September 2011
7. Zhou, M., Tang, Y.: GrassMA: graph-based semi-supervised manifold alignment for indoor WLAN localization. *IEEE Sens. J.* <https://doi.org/10.1109/jsen.2017.2752844>
8. Zhang, L., Ma, L., Xu, Y.: A semi-supervised indoor localization method based on 11-graph algorithm. *J. Harbin Inst. Technol. (New Series)* **22**(4), 55–61 (2015)

9. Wang, H., Sen, S., Elgohary, A., Farid, M., Youssef, M., Choudhury, R.R.: No need to war-drive: Unsupervised indoor localization. In: Proceedings of the 10th International Conference on Mobile Systems, Applications, and Services, pp. 197–210 (2012)
10. Unsupervised learning for crowdsourced indoor localization in wireless networks. *IEEE Trans. Mob. Comput.* **15**(11) (2016)
11. Borg, I., Groenen, P.: *Modern Multidimensional Scaling: Theory and Applications*, 2nd edn., pp. 207–212. Springer, New York (2005). ISBN 0-387-94845-7
12. Yang, C., Shao, H.-R.: WiFi-based indoor positioning. *IEEE Commun. Mag.* **53**, 150–157 (2015)
13. Ji, Z., Li, B.-H., Wang, H.-X.: A new indoor ray-tracing propagation prediction model. In: *Computational Electromagnetics and Its Applications*



# Achieving Multiagent Coordination Through CALA-rFMQ Learning in Continuous Action Space

Wanshu Liu<sup>1</sup>, Chengwei Zhang<sup>2</sup>, Tianpei Yang<sup>1</sup>, Jianye Hao<sup>1</sup>(✉),  
Xiaohong Li<sup>2</sup>, and Zhijie Bao<sup>3</sup>

<sup>1</sup> School of Computer Software, Tianjin University, Tianjin, China  
liuwanshu2017@hotmail.com, {jianye.hao, tpyang}@tju.edu.cn

<sup>2</sup> School of Computer Science and Technology, Tianjin University, Tianjin, China  
{cheny, xiaohongli}@tju.edu.cn

<sup>3</sup> School of Textiles, Tianjin Polytechnic University, Tianjin, China  
flmcowdit@outlook.com

**Abstract.** In cooperative multiagent systems, an agent often needs to coordinate with other agents to optimize both individual and system-level payoffs. A lot of multiagent learning approaches have been proposed to address coordination problems in discrete-action cooperative environments. However, it becomes more challenging when faced with continuous action spaces, e.g., slow convergence rate and convergence to suboptimal policy. In this paper, we propose a novel algorithm called CALA-rFMQ (Continuous Action Learning Automata with recursive Frequency Maximum Q-Value) that ensures robust and efficient coordination among multiple agents in continuous action spaces. Experimental results show that CALA-rFMQ facilitates efficient coordination, and outperforms previous works.

**Keywords:** Multiagent learning · Coordination games  
Continuous action space

## 1 Introduction

In multiagent systems, an agent often needs to coordinate with other agents. Nowadays, significant efforts have been devoted to multiagent coordination problems. Most of these approaches are extended from Q-learning, such as distributed Q-learning [8,9], the hysteretic Q-learning [5], and recursive Frequency Maximum Q-Value (rFMQ) [5,6]. However, these algorithms can only handle the multiagent coordination problem in discrete action spaces.

A number of researches have been devoted to dealing with the single-agent learning problems in continuous action spaces. The most common algorithms are based on function approximation technology [1,11–16]. The others are based on the Monte Carlo sampling method [17,18]. All these algorithms have problems



of slower convergence or finding the local optimum actions rather than global optimum in continuous action spaces. Furthermore, all of them are designed for single-agent environment and cannot be applied in multiagent systems directly. One notable exception is proposed by De Jong et al. [19] which extended the Continuous Action Learning Automata (CALA) algorithm from single-agent domains to multiagent domains, but this method aims at achieving fairness rather than cooperation in multiagent continuous action spaces.

In this paper, we propose a framework to transfer the prior knowledge from discrete-action learning algorithms to continuous action space to improve the performance of continuous action algorithms. According to this framework, we propose a novel algorithm called Continuous Action Learning Automata with recursive Frequency Maximum Q-Value (CALA-rFMQ) that leverages the advantage of existing multiagent discrete-action learning algorithms and single-agent continuous action algorithms. The core idea of CALA-rFMQ is that agent divide the original continuous action space into several sub-continuous action spaces and transfer the knowledge obtained by a discrete learning algorithm to a continuous learning algorithm, and then explores the optimal action in sub-action spaces.

The rest of the paper is organized as follows. Section 2 introduces the basic algorithms that conclude the CALA and the rFMQ. The detailed description of CALA-rFMQ is presented in Sect. 3. The benefits of our algorithm are experimentally demonstrated in Sect. 4. Section 4 concludes and points out new directions.

## 2 Preliminaries

### 2.1 CALA

CALA [1, 3, 19] is a learning automata developed for problems with continuous action spaces. CALA enables each agent to establish and maintain a Gaussian distribution  $N(\mu, \sigma^2)$ . At each episode, each agent updates its action probability distribution based on its interaction with the environment by updating  $\mu$  and  $\sigma$ . CALA agent needs to take two actions  $x$  and  $\mu$ , and gets two feedbacks  $V(x)$  and  $V(\mu)$  from the environment in each step respectively where  $\mu$  is the action corresponding to the mean of Gaussian distribution, and  $x$  is the action sampled from the Gaussian distribution  $N(\mu, \sigma^2)$ . Then, CALA uses  $V(x)$  and  $V(\mu)$  to update the value of  $\mu$  and  $\sigma$  as follows,

$$\mu = \mu + \alpha \frac{V(x) - V(\mu)}{\phi(\sigma)} \frac{x - \mu}{\phi(\sigma)}. \quad (1)$$

$$\sigma = \sigma + \alpha \frac{V(x) - V(\mu)}{\phi(\sigma)} \left[ \left( \frac{x - \mu}{\phi(\sigma)} \right)^2 - 1 \right] - \alpha K(\sigma - \sigma_L). \quad (2)$$

$$\phi(\sigma) = \max(\sigma, \sigma_L) \quad (3)$$

where  $\alpha$  is the learning rate, and  $K$  represents a large positive constant driving down  $\sigma$ . The iterations ends when the  $\sigma$  is dropped to the threshold  $\sigma_L$ , and  $\mu$  does not change dramatically.

## 2.2 rFMQ

The rFMQ is an improved version of FMQ [4] proposed by Matignon [5]. Each agent  $i$  in rFMQ holds the ordinary  $Q$ -value  $Q_i(a)$ , the maximum reward  $Q_{i,max}(a)$  and  $E$ -value  $E_i(a)$  for each action  $a$ . The key idea of rFMQ is to evaluate the actions using a linear interpolation based on the occurrence frequencies. Actions evaluations fluctuate between optimistic and mean evaluation according to the stochasticity of the game.  $E$ -value  $E_i(a)$  is updated as follows,

$$E_i(a) \leftarrow (1 - F_i(a))Q_i(a) + F_i(a)Q_{i,max}(a) \quad (4)$$

The frequency  $F_i(a)$  is used to evaluate the frequency of receiving the maximum reward  $Q_{i,max}(a)$  when agent  $i$  plays action  $a$ , and it is recursively computed using a learning rate  $\alpha_f$ . Formally, it has,

$$F_i(a) \leftarrow \begin{cases} 1 & r > Q_{i,max}(a) \\ (1 - \alpha_f)F_i(a) + \alpha_f & r = Q_{i,max}(a) \\ (1 - \alpha_f)F_i(a) & r < Q_{i,max}(a) \end{cases} \quad (5)$$

where  $r$  is the reward received in the current round. However the main disadvantage of rFMQ is that it is only suitable for matrix games [5,6].

## 3 CALA-rFMQ

Note that although we present a single-state algorithm description, CALA-rFMQ can be extended to multiple states. Inspired from the idea of decision making in CALA and coordination method in rFMQ, we propose a novel algorithm CALA-rFMQ to address the coordination problems of multiagent in continuous action spaces. The key idea of CALA-rFMQ is to divide the continuous action spaces into sub-continuous action spaces by actions which were selected by discrete learning algorithm and explore the optimal action from the sub-continuous action spaces based on continuous learning algorithm. The overall framework of the CALA-rFMQ learning algorithm is presented in Algorithm 1.

In each round, each agent  $i$  firstly samples  $N$  discrete actions evenly from continuous action space (Line 5). Secondly, CALA-rFMQ agent iteratively selects the top  $k$  actions from the  $N$  sampled actions (Line 6–8), which is shown in Algorithm 2. Combining with the idea of rFMQ learning, we extend the Policy Hill-Climbing (PHC) algorithm to cooperative multiagent games. PHC is a simple rational learning algorithm that is capable of playing mixed strategies in multiagent environments. The reason why we select top  $k$  actions instead of only one optimal action is that one chosen action from the discrete samples may not be the best one, and choose top  $k$  actions ensures that the transferred knowledge

**Algorithm 1.** CALA-rFMQ

---

```

1: Initialization: the number of samples  $N$ , the number of selected actions  $k$ , the
   sample interval  $m$ , the threshold  $\eta$ , the continuous action set  $\mathbf{A}_i$ 
2:  $\forall a \in A_i, Q_i(a) \leftarrow 0, Q_{i,max}(a) \leftarrow 0, F_i(a) \leftarrow 1, E_i(a) \leftarrow 0, \pi_i \leftarrow \frac{1}{|A_i|},$ 
    $\sigma_i, \alpha_w, \alpha_l, \alpha, \alpha_{cmax_i}, \alpha_{cmin_i}$ 
3: for each episode do
4:   for each agent  $i$  do
5:     Discretization: Get  $N$  samples uniformly from  $\mathbf{A}_i$  at intervals of  $m$ .
6:     repeat
7:       Get the top  $k$  optimal actions from  $N$  samples (see Algorithm 2.)
8:     until  $\sum \pi_{i_k} \geq \eta$ 
9:     Find the final optimal action with prior knowledge (see Algorithm 3).
10:   end for
11: end for

```

---

**Algorithm 2.** rFMQ with PHC

---

```

1: repeat
2:   for  $episode \leq k$  do
3:     Agent  $i$  selects  $a_i$  following the policy  $\pi_i$  and with some exploration.
4:     Apply joint action  $a$  and observe reward  $r$ 
5:     Update  $Q_i(a) \leftarrow (1 - \alpha)Q_i(a) + \alpha r$ 
6:     Update  $E_i(a)$  and  $F_i(a)$  according to the Eqs. (4) and (5).
7:     Update  $\pi_i$  according to  $E_i(a)$ 
8:     if  $a \neq argmax E_i(a)$  then
9:        $\pi_i(a) \leftarrow \pi_i(a) + \frac{-\delta}{|A_i|-1}$ 
10:    else
11:       $\pi_i(a) \leftarrow \pi_i(a) + \delta$ 
12:    end if
13:   end for
14: until  $\sum \pi_{i_k} \geq \eta$ 

```

---

from the discrete samples is effective. The selection ends when the sum of the policies  $\pi$  corresponding to top  $k$  actions is above the given threshold  $\eta$ .

After obtaining top  $k$  actions, CALA-rFMQ agent divides its continuous action space into  $k$  sub-spaces. Then, each agent  $i$  transfers the top  $k$  actions, the  $E$ -values and the policies of these actions to the continuous algorithm as initializations. At last, drawing on the idea of Win or Learn Fast (WoLF) principle [2], we propose Win or Learn Slowly (WoLS) principle and combine it with the idea of CALA to explore the final optimal action from the  $k$  sub-continuous action spaces with the prior knowledge, which is presented in Algorithm 3 (Line 9). WoLS is diametrically opposite to the WoLF principle, for the reason that WoLF aims at guaranteeing the convergence of the algorithm, while WoLS is designed to correctly estimate the expected payoff of action, and guarantee the update of action in the direction of increasing the agents payoff.

### 3.1 Optimum Discrete Actions Using rFMQ with PHC

Algorithm 2 demonstrates how CALA-rFMQ extends the idea of rFMQ with PHC to obtain top  $k$  actions in the discrete-action spaces. Each agent  $i$  selects an action following its policy  $\pi_i$  according to  $\epsilon$ -greedy mechanism from the  $N$  sampled actions and plays the game with other agent to observe the reward of the joint action  $\{a_0, a_1 \dots a_i\}$  (Line 3–4). After that, for each agent, CALA-rFMQ uses the rFMQ to calculate the  $E$ -value and frequency of the selected action (Line 5–6). Finally, CALA-rFMQ uses the PHC to update the policy  $\pi_i$  for each action in the  $N$  discrete samples according to  $E$ -value (Line 7–12). This process repeats iteratively for  $k$  rounds. The selection ends when the sum of the policies  $\pi_i$  of top  $k$  actions is greater than a given threshold  $\eta$  (Line 14).

### 3.2 Win or Learn Slow Continuous Action Learning Automata (WoLS-CALA)

WoLS-CALA is shown in Algorithm 3. In each round, for each agent  $i$ , CALA-rFMQ reuses the top  $k$  actions, and the  $E$ -values and the policies  $\pi$  of these actions as initial means  $\mu$ , the corresponding values  $V(\mu)$  of the initial mean and the corresponding policies  $\pi_\mu$  of the initial mean respectively (Line 1). We first project the obtained policy  $\pi_{i_k}$  to the  $[0,1]$  interval (Line 2). Secondly, for each agent the continuous action spaces are divided into  $k$  sub-continuous action spaces according to  $k$  selected actions. In each sub-continuous action spaces, the choice of action, the update of  $\mu_{i_k}$  and  $\sigma_{i_k}$  are depending on  $V(x)$  and  $V(\mu)$ .

---

#### Algorithm 3. WoLS-CALA with prior knowledge

---

- 1: Get initial  $\mu_{i_k}$ ,  $V(\mu_{i_k})$  and  $\pi_{i_k}$  for each agent  $i$  from Algorithm 2
  - 2: Project the policy  $\pi_{i_k}$  to the  $[0,1]$  interval.
  - 3: **for** each action in top  $k$  actions **do**
  - 4:   Get  $x_{i_k} \sim N(\mu_{i_k}, \sigma_{i_k}^2)$  and expected value  $V(x_{i_k})$
  - 5:   **if**  $V(x_{i_k}) > V(\mu_{i_k})$  **then**
  - 6:      $a_{i_k} = x_{i_k}$
  - 7:     Update  $\mu_{i_k}$  and  $\sigma_{i_k}$  using **Eqs. (1), (2) and (3)** with Wining
  - 8:     learning rate  $\alpha_w$
  - 9:   **else**
  - 10:     $a_{i_k} = \mu_{i_k}$
  - 11:    Update  $\mu_{i_k}$  and  $\sigma_{i_k}$  using **Eqs. (1), (2) and (3)** with Losing
  - 12:    learning rate  $\alpha_l$
  - 13:   **end if**
  - 14:   Put action  $a_{i_k}$  into action set  $A_{c_i}$
  - 15: **end for**
  - 16: Get final optimal action  $a_{i_t}$  from  $A_{c_i}$  according to  $\pi_{i_k}$  with exploration.
  - 17: Get final reward  $r$  according to final joint action  $a_t$
  - 18: Update  $V(\mu_{i_k}) \leftarrow V(\mu_{i_k}) + \alpha r$
  - 19: Get reward  $r_k$  according to joint action  $a_k$  and put it into CALA reward set  $R_{a_c}$ .
  - 20: Update  $\pi_{i_k}$  according to the **Eq. (6)**
-

A larger  $V(x)$  means  $x$  is better action, so WoLS-CALA agent chooses  $x$  as action  $a_{i_k}$ ; otherwise it chooses  $\mu$ . WoLS-CALA agent uses a larger learning rate  $\alpha_w$  to change quickly if the received payoff  $V(x)$  of action  $x$  is greater than the current mean value  $V(\mu)$  of the mean action  $\mu$  (winning), otherwise it uses a smaller learning rate  $\alpha_l$  (losing) (Line 5–13). It can guarantee that the actions update is in the direction of increasing the agents accumulated rewards.

In the end of exploration, each CALA-rFMQ agent obtains  $k$  actions and puts these actions into its CALA-action set  $A_{c_i}$  (Line 14). According to policy  $\pi_{i_k}$  with some exploration, each CALA-rFMQ agent obtains final optimal action  $a_{t_i}$  from  $A_{c_i}$  (Line 16). After playing game with other agents, the final reward of final joint action  $a_t$  is decided (Line 17), which is used to update the value of the mean for each agent  $i$  (Line 18). In addition, the reward  $r_k$  of joint action  $a_k$  in each sub-continuous action spaces make up a CALA-reward set  $R_{a_c}$ , which is used to update the policies of each agent (Line 19–20). According to CALA-action set  $A_{c_i}$  and CALA-reward set  $R_{a_c}$ , we can obtain the action  $a_{cmax_i}$  with maximum reward. The policies of each action is updated as follow.

$$\pi_{i_k} \leftarrow \begin{cases} \pi_{i_k} \leftarrow \pi_{i_k} + \alpha_{cmax_i} & a_{i_k} = a_{i_t} \& a_{i_k} = a_{cmax_i} \quad (a) \\ \pi_{i_k} \leftarrow \pi_{i_k} - \alpha_{cmax_i} & a_{i_k} = a_{i_t} \& a_{i_k} \neq a_{cmax_i} \quad (b) \\ \pi_{i_k} \leftarrow \pi_{i_k} + \alpha_{cmin_i} & a_{i_k} \neq a_{i_t} \& a_{i_k} = a_{cmax_i} \quad (c) \\ \pi_{i_k} \leftarrow \pi_{i_k} - \alpha_{cmin_i} & a_{i_k} \neq a_{i_t} \& a_{i_k} \neq a_{cmax_i} \quad (d) \end{cases} \quad (6)$$

For each action  $a_{i_k}$  of each agent in CALA-action set  $A_{c_i}$ , if  $a_{i_k}$  is the maximum action  $a_{cmax_i}$ , the probability  $\pi_{i_k}$  of choosing  $a_{i_k}$  is increased, otherwise it is decreased. Among them, if action  $a_{i_k}$  is the final action  $a_{i_t}$ , the policy  $\pi_{i_k}$  increases or decreases with a greater learning rate  $\alpha_{cmax_i}$ ; otherwise it increases or decreases with a small learning rate  $\alpha_{cmin_i}$ . At last all the policies  $\pi_{i_k}$  should be projected to the  $[0, 1]$  interval.

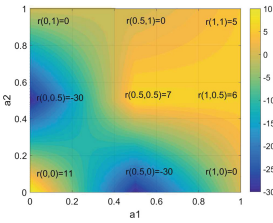
## 4 Experimental and Conclusion

In this section, we evaluate the performance of CALA-rFMQ compared with previous work [1] and the influence of key parameters on the representative Fully Climbing Game (FCG) [7] and the Partially Stochastic Climbing Game (PSCG) [6–8] which are cooperative matrix game and presented in Table 1(a) and (b). It is obvious that in FCG and PSCG, the joint action  $(a, a)$  is the pareto-dominate Nash equilibrium and that  $(b, b)$  is a suboptimal Nash equilibrium. Using the bilinear interpolation techniques, we construct continuous action game models as shown in Fig. 1.

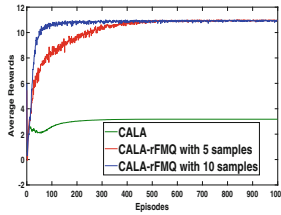
Figures 2 and 3 first show the results of comparison of CALA-rFMQ with CALA in continuous action version of FCG and PSCG respectively. We use the average reward of 500 runs to evaluate the performance of CALA-rFMQ and CALA. As we can see in the Figs. 2 and 3, CALA-rFMQ converges to the joint action  $(a, a)$  faster and gets average payoff of 11. However, CALA only finds the local optimal action that is closest to the initial mean. Therefore, CALA-rFMQ outperforms CALA in terms of dealing with the coordination problem in

**Table 1.** The climbing game

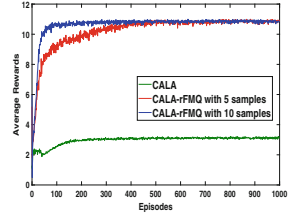
		(a) FCG			(b) PSCG				
		Agent 1's actions			Agent 1's actions				
		a	b	c					
Agent 2's actions	a	11	-30	0	Agent 2's actions	a	11	-30	0
	b	-30	7	6		b	-30	14/0	6
	c	0	0	5		c	0	0	5



**Fig. 1.** The color map of the continuous FCG (Color figure online)



**Fig. 2.** FCG



**Fig. 3.** PSCG

single-stage games with continuous action space. Furthermore, the more samples CALA-rFMQ agent obtains from discrete-action spaces, the better performance it achieves as shown in Figs. 2 and 3. The reason is that if the number of samples obtained by uniform discretization of CALA-rFMQ is more, CALA-rFMQ agent has a greater probability to choose near the optimal actions in the discrete-action spaces, which is more effective for continuous learning algorithm.

In this paper, we propose a framework that transfers the knowledge of discrete-action learning algorithms to continuous action space to improve the performance of continuous action algorithms. According to this framework, we propose CALA-rFMQ algorithm to address the coordination problem in the continuous action cooperative games, and the experiments show that CALA-rFMQ outperforms other multiagent learning algorithms in terms of dealing with the coordination problems. In the future, we first extend our experiment to multiple states. In addition, we will use other discrete and continuous multiagent learning algorithms to verify the effectiveness of our framework.

**Acknowledgments.** The work is supported by the National Natural Science Foundation of China under Grant No.: 61702362 and Special Program of Artificial Intelligence of Tianjin Municipal Science and Technology Commission (No.:569 17ZXRGX00150).

## References

1. Thathachar, M., Sastry, P.: *Networks of Learning Automata: Techniques for Online Stochastic Optimization*. Kluwer Academic Publishers, Boston (2004)
2. Bowling, M., Veloso, M.: Multiagent learning using a variable learning rate. *Artif. Intell.* **136**(2), 215–250 (2002)
3. Tuyls, K., Now, A.: Evolutionary game theory and multi-agent reinforcement learning. *Knowl. Eng. Rev.* **20**(1), 63–90 (2005)
4. Kapetanakis, S., Kudenko, D.: Reinforcement learning of coordination in cooperative multi-agent systems. In: *AAAI/IAAI*, pp. 326–331 (2002)
5. Matignon, L., Laurent, G.J., Le Fort-Piat, N.: Independent reinforcement learners in cooperative markov games: a survey regarding coordination problems. *Knowl. Eng. Rev.* **27**(1), 1–31 (2012)
6. Hao, J., Huang, D., Cai, Y., et al.: The dynamics of reinforcement social learning in networked cooperative multiagent systems. *Eng. Appl. Artif. Intell.* **58**, 111–122 (2017)
7. Claus, C., Boutilier, C.: The dynamics of reinforcement learning in cooperative multiagent systems. In: *Proceedings of the Fifteenth National Conference on Artificial Intelligence*, pp. 746–752 (1998)
8. Lauer, M., Riedmiller, M.: An algorithm for distributed reinforcement learning in cooperative multi-agent systems. In: *Proceedings of the Seventeenth International Conference on Machine Learning* (2000)
9. Chen, X., Duan, Y., Houthoofd, R., et al.: Infogan: interpretable representation learning by information maximizing generative adversarial nets. In: *Advances in Neural Information Processing Systems*, pp. 2172–2180 (2016)
10. Ray, S.S.: *Numerical Analysis with Algorithms and Programming*. CRC Press, Boca Raton (2016)
11. Alibekov, E., Kubalk, J., Babuka, R.: Policy derivation methods for critic-only reinforcement learning in continuous spaces. *Eng. Appl. Artif. Intell.* **69**, 178–187 (2018)
12. Sutton, R.S., Maei, H.R., Precup, D., et al.: Fast gradient-descent methods for temporal-difference learning with linear function approximation. In: *Proceedings of the 26th Annual International Conference on Machine Learning*, pp. 993–1000. ACM (2009)
13. Galstyan, A.: Continuous strategy replicator dynamics for multi-agent Q-learning. *Auton. Agents Multi-Agent Syst.* **26**(1), 37–53 (2013)
14. Lillicrap, T.P., Hunt, J.J., Pritzel, A., et al.: Continuous control with deep reinforcement learning. arXiv preprint [arXiv:1509.02971](https://arxiv.org/abs/1509.02971) (2015)
15. Peters, J., Schaal, S.: Reinforcement learning of motor skills with policy gradients. *Neural Netw.* **21**(4), 682–697 (2008)
16. Van Hasselt, H.: Reinforcement learning in continuous state and action spaces. In: Wiering, M., van Otterlo, M. (eds.) *Reinforcement Learning*, pp. 207–251. Springer, Heidelberg (2012). [https://doi.org/10.1007/978-3-642-27645-3\\_7](https://doi.org/10.1007/978-3-642-27645-3_7)
17. Van Hasselt, H., Guez, A., Silver, D.: Deep reinforcement learning with double Q-Learning. In: *AAAI 2016*, pp. 2094–2100 (2016)
18. Lazaric, A., Restelli, M., Bonarini, A.: Reinforcement learning in continuous action spaces through sequential Monte Carlo methods. In: *Advances in Neural Information Processing Systems*, pp. 833–840 (2008)
19. De Jong, S., Tuyls, K., Verbeeck, K.: Artificial agents learning human fairness. In: *Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS 2008*, vol. 2, pp. 863–870 (2008)



# Environmental Reconstruction for Autonomous Vehicle Based on Image Feature Matching Constraint and Score

Fangchao Hu<sup>(✉)</sup> , Ling Bai , Yinguo Li , and Zhen Tian 

Chongqing University of Posts and Telecommunications, Chongqing 400065, China  
fangchaohu1211@126.com

**Abstract.** The environment perception of autonomous vehicle is mainly aimed at obtaining the holonomic information around the ego-vehicle to understand the driving environment. In this paper, we attempt to use only vehicle-mounted cameras to reconstruct the 3D environment for the autonomous vehicle perception. Firstly, we acquire the continuous frames in vehicle motion to reconstruct continuous 3D point clouds. Secondly, the continuous point clouds can be stitched by the proposed matching constraint and scores of image features. Lastly, we get high-accuracy and high-efficiency dense point cloud of the environment. Experimental results on the benchmark dataset demonstrate the effectiveness and robustness of the proposed stitching algorithm. Compared with other algorithms and its variants, the MSE of the proposed method is lower than the average and the number of redundant points in overlap regions is reduced.

**Keywords:** 3D point cloud · Image feature matching constraint  
Point cloud stitching

## 1 Introduction

The understanding of the scene around the vehicle is the prerequisite for the proper operation of automatic driving. The cameras are mounted on the autonomous vehicle to perceive the environment. The viewing angle of the mounted cameras are usually not wide, so the picture does not encompass the entire scene with a single image. Therefore, many pictures must be taken to obtain a wide-angle scene around the autonomous vehicle when the vehicle is moving. Then, the pictures from the vehicle's mounted camera are stitched together to create a complete image. Most of the 2D algorithm stitched images have achieved perfect performance without obvious seams [11]. Panoramic mosaic of 2D images cannot fulfil the application requirements of autonomous vehicle environment perception. For example, the information of distance and the actual size of the objects are unknowable in a 2D image. The distance between the objects and the autonomous vehicle is essential for the control stage. Therefore, 3D point cloud stitching that contains the object's location information may satisfy the requirement. However, 3D point cloud stitching has problems related to accuracy and instantaneity due to its



unstructured and time-consuming processing. In this paper, our goal is to improve the speed and accuracy of stitching 3D point clouds.

In view of the 3D scene reconstruction that the vehicle can understand and the environmental mapping that allows the vehicle to analyze the trend of the target movement, the 3D point cloud stitching is a crucial step in environmental perception. 3D point cloud stitching has many applications, such as rebuilding indoor scenes to allow autonomous robotic platforms to achieve sophisticated manipulation capabilities, building the 3D models from photos taken with a handheld camera, reconstructing interest points that are absent for architectural modeling, automating progress tracing, etc. The mainstream methods of 3D point cloud stitching are classified into three categories: the template matching methods [1, 6, 8], feature matching methods [3, 5, 10, 11], and voxel matching methods [4, 7].

The scene in this work is captured by moving cameras, and the objects in this scenario are large, but the objects may be confused with the background. To address these problems, a feature matching constrains and score-based method is proposed. We deployed two methods to improve the accuracy and speed. First, the idea of the feature matching score-based stitching method is to stitch the continuous point cloud according to the score of matching features. Second, the constrain of feature tracking between temporal frames in 2D image sequence is to obtain the transformation matrix for point cloud stitching.

## 2 Proposed Framework

We investigated two algorithms for matching pairs of acquired 3D point clouds: ICP (Iteration Closest Points) [8, 9] and NDT (Normal Distributions Transform) [5]. ICP is the actual standard 3D registration algorithm which is used in the 3D stitching community, and NDT has appeared to be a compelling alternative in previous comparisons. The purpose of ICP is minimizing the Eq. (1)

$$E(R, t) = \sum_{i=1}^{N_m} \sum_{j=1}^{N_d} \omega_{i,j} \left\| m_i - (Rd_j + t) \right\|^2 \quad (1)$$

Wherein  $N_m$  and  $N_d$  are the number of points in the model set  $M$  and data set  $D$ , respectively.  $\omega_{i,j}$  are the weights for a point match.

In this paper, the main task is to obtain the multi-view information from the mounted stereo camera by using point cloud stitching. The single view scene can be reconstructed by the associated features within two images. With the vehicle moving, we can get the multi-view single point clouds. The matching constrain and score-based method are proposed to stitch the point clouds time-saving and accurately. This matching constrains and scores affect the density and accuracy of the stitched point clouds. Each of the candidate point clouds will get a score, then each score is normalized as a stitching weigh. The weight can control the number of point clouds which are in the overlap region. With the features moving between the adjacent frames, we track the features to

obtain transformation matrix like that in Eq. (1). We find the minimum reprojection error, as shown in Eq. (2).

$$T_{k,k-1} = \arg \min_T \sum_i \|u'_i - \pi(p_i)\|_{\Sigma}^2 \quad (2)$$

Wherein the  $T_{k,k-1}$  is the transformation matrix from frame  $k-1$  to  $k$ ,  $u'_i$  is the feature point in frame  $k$ (actual value). The  $\pi(p_i)$  is the reprojection point of  $p_i$  in frame  $k$ (theoretical value). We consider this constrain, we modified the objective function as follow to obtain optimized transformation matrix.

$$E(R, t) = \sum_{i=1}^{N_m} \sum_{j=1}^{N_d} \|m_i - (Rd_j + t)\|^2 + \sum_i \|u'_i - K \times [R|t] \times p_i\|_{\Sigma}^2 \quad (3)$$

Wherein the  $K$  is the intrinsic parameter of camera. The combined constrain not only reduce the mismatching but also improve the calculate speed, due to the limited feature pairs.

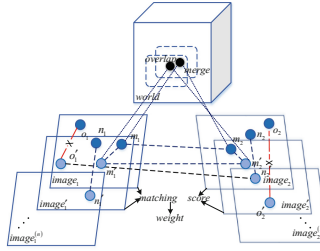
Features detecting and matching of the image pairs are the essential issue to produce the scores. Assuming that the numbers of features of each image pair are  $C_{n1}$ ,  $C_{n2}$  ( $n = 1 \dots N$ ), respectively, wherein  $n$  is the sequence number of frame. Then the number of matched features is  $\min(\cdot)$ , the scores of each image pair can be produced as followed:

$$score_n = \frac{\min(C_{n1}, C_{n2})}{\sum_{n=1}^N \min(C_{n1}, C_{n2})} (n = 1, 2, \dots, N) \quad (4)$$

In other words, the scores equal to the sum of the minimum between  $C_{n1}$  and  $C_{n2}$  of each frame divided by the number of matched features  $\min(\cdot)$  of corresponding frame. And then the scores will be normalized into the weight:

$$weight_{n+1} = \frac{score_{n+1}}{score_n + score_{n+1}} (n = 1, 2, \dots, N) \quad (5)$$

The weight is gained by the two adjacent frames scores, theoretically, the overlap merely occurs at adjacent frames. Therefore, we should only compute the weight between two adjacent frames. In Fig. 1, the feature points  $n_1$ ,  $n'_1$  are matched in the adjacent frames  $image_1$ ,  $image'_1$ , the matching score are calculated according to the Eq. (4). Then the weight is gained according to the Eq. (5).  $o_1$ ,  $o'_1$  are linked with red dashed line which indicate the mismatch points, between the adjacent frames  $image_1$ ,  $image'_1$ . The matching score can be calculated by these matched feature points, and then the weights between the adjacent frames can be obtained.



**Fig. 1.** Principle of the matching score

The point clouds are stitched by the score-based algorithm, as Algorithm 1 stated. The number of point clouds are multiplied by a weight which is produced by the scores of adjacent frames, which makes the mosaicked point clouds more precise. Because in the presented algorithm, the points will be multiplied by the weight, then the points in the overlap will be reduced correspondingly. The algorithm of weight generation of adjacent frames satisfied the following equations.

$$N_A + N_B = N_C \tag{6}$$

$N_A, N_B$  are the number of points in the adjacent point clouds,  $N_C$  is the sum of points in the two adjacent point clouds.

$$weight_\alpha \cdot N_A + weight_\beta \cdot N_B = 1/2N_C \tag{7}$$

Equation (6) into Eq. (7), we can get.

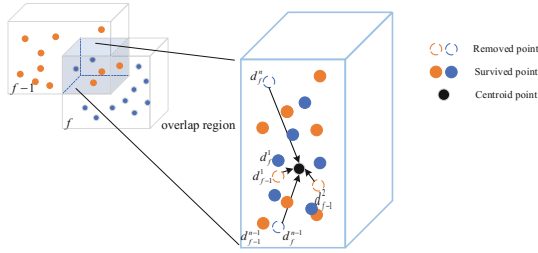
$$weight_\alpha = \frac{\left(\frac{1}{2} - weight_\beta\right)N_B}{N_A} + \frac{1}{2} \tag{8}$$

From the Eq. (7) we can get.

$$weight_\alpha + weight_\beta = 1 \tag{9}$$

$weight_\alpha, weight_\beta$  are the weights calculated by the matching scores between the two adjacent point clouds.

As shown in Fig. 2, the blue cubic 3D box represents the overlap region between two adjacent point clouds. The black solid point is the centroid point among the points in the overlap region. The orange points denote the points in frame  $f - 1$ . The blue points denote the points in frame  $f$ .  $d_f^n$  denotes the point is in the frame  $f$  and is the  $n$  point of nearest distance to the centroid point.  $d_{f-1}^1$  denotes the point is in the frame  $f - 1$  and is the first point with the nearest distance to the centroid point. The solid points are surviving points, and the hollow points are removed points.



**Fig. 2.** Points reduction rule of end to end. (Color figure online)

The redundancy points will be reduced compare to the conventional algorithm. And the number of points is controlled according to the scores, which are relevant to the performance of features matching.

---

**Algorithm 1. Matching Score based Point cloud stitch algorithm**

---

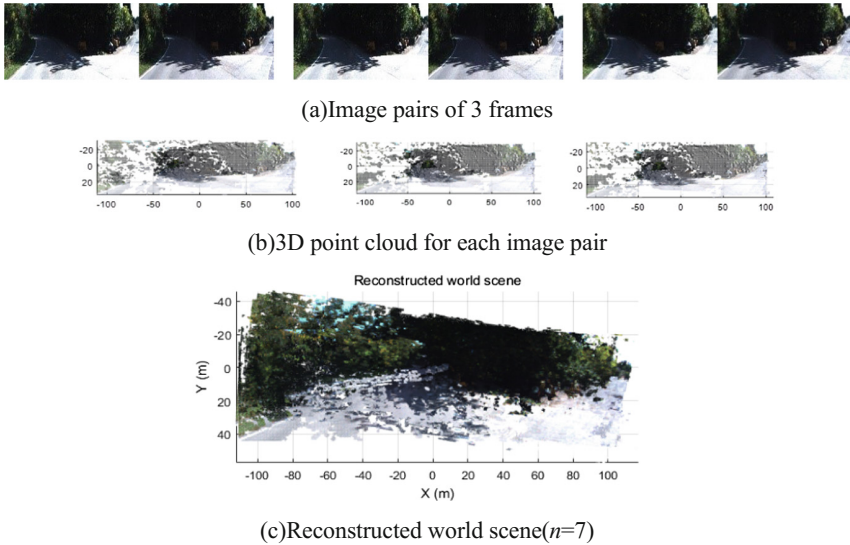
```

Input: Continuous Point Cloud  $P_n(0 < n < N)$ 
          Initialized Point Cloud  $C$ 
Output: Stitched Point Cloud  $\{C_m\}$  ( $0 < m < M$ )
While ( $P_n \neq \phi$ )
    IF  $P_n(0 < n < N) \in$  moving point cloud
      IF  $P_n(0 < n < N) \notin$  limit region
        The  $P_n$  was added to the point cloud  $C$ 
         $\rightarrow P_n + \sum\{C_N\}$ 
      ELSE
        Calculate the weight according to the match score  $\rightarrow$ 
         $weight_\alpha$ 
      Then the weight multiplied by  $P_n \rightarrow weight_\alpha \cdot P_n$ 
      The weighted  $P_n$  was added to the point cloud  $C$ 
       $\rightarrow weight_\alpha \cdot P_n + \sum\{C_N\}$ 
      IF END
    ELSE
      Cast off this point
    IF END
     $n = n + 1$ 
END
  
```

---

### 3 Experiment

The experimental data set of this work consists of two parts, one is the 1385 image pairs captured by a calibrated binocular camera composed of two monocular cameras (Basler ace-acA2500-60uc), and the other is 1594 image pairs in the challenging public data set KITTI [2]. The experimental environment platform for this paper is: C++, MATLAB 2016(a), and Intel Core i7-4700 CPU@2.4 GHz with 8G RAM. As shown in Fig. 3(a), the 3D world scene with  $n$  frames of 3D point cloud is reconstructed by the proposed method in this paper. The metric information of the world is given in Fig. 3(c). We set the location of left camera as the origin of coordinates, right and down are the plus direction. Figure 3(a) is the 3 frames original image pairs and Fig. 3(b) is the reconstructed 3D point cloud for each image pair in Fig. 3(a). We can see that the scene reconstructed with dense 3D point cloud in this paper is approximate to the real.



**Fig. 3.** Reconstructed 3D point cloud of world scene

In order to get the accurate point cloud reconstruction, meanwhile to get higher score of matching, several frequently-used methods of feature detection and matching are experimented to find the highest match score. As shown in Table 1, we deployed 6 type of feature detection algorithm to calculate score and weight.

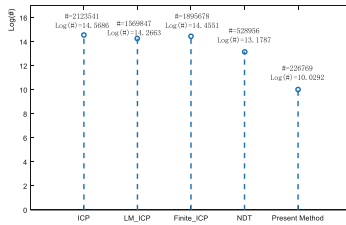
As can be seen in Table 2, the proposed method in this paper has low time consumption compared with other algorithms because it considers the mismatch problem in overlapping region. Furthermore, we can find in Fig. 4 that the proposed method obtains the minimum number of points after stitching than other methods.

**Table 1.** Scores of different algorithms of feature detection with image pair sequence

Feature detection	Minimum number of features	Number of matched	Score	Weight
<i>SURF</i>	5611	<b>1332</b>	<b>0.2374</b>	<b>0.3048</b>
<i>FAST</i>	5379	660	0.1227	0.1576
<i>MinEigen</i>	<b>14046</b>	948	0.0675	0.0867
<i>Harris</i>	4413	540	0.1224	0.1572
<i>BRISK</i>	3048	109	0.0358	0.0460
<i>MSER</i>	4326	835	0.1930	0.2478

**Table 2.** Time costing in different algorithm of stitching

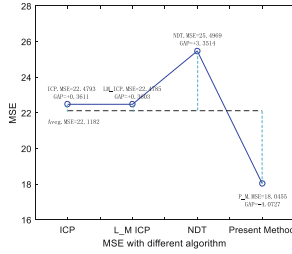
Algorithm of point cloud stitch	Time cost of computing (s)
<i>ICP</i>	0.794970
<i>L-M ICP</i>	0.936874
<i>NDT</i>	1.200785
<i>Present method</i>	0.661914



**Fig. 4.** Total number of stitched point cloud with different algorithm

Figure 5 shows the MSE (Mean Square Error) of different algorithms and gaps to the average MSE. MSE can reflect the difference between estimated and the benchmark data. MSE calculated are as follows in Eq. (10). The smaller the MSE, the higher the accuracy of the reconstructed 3D point cloud. It can be seen from Fig. 5 that the proposed algorithm has the lowest MSE, and the proposed is 23.13% lower than the average MSE as the other methods is higher than the average MSE.

$$MSE = \sqrt{\frac{\sum_{i=1}^N (X_{obs,i} - X_{model,i})^2}{n}} \tag{10}$$



**Fig. 5.** MSE of different algorithm and gaps to the average MSE

From the experimental data and figures, we can see that the proposed method, which can reconstruct the scene using point cloud, achieves the goal of point cloud stitching sufficiently without obvious seam. In addition, the proposed method reduces a large number of redundant points which are unnecessary to process in the overlap region. The occupancy of store space is effectively reduced and the accumulate errors can cut down because of the deletion of the point cloud with low matching score.

## 4 Conclusion

This paper proposed a novel stitching algorithm of 3D point cloud for autonomous vehicle to reconstruct the 3D environment. The proposed matching constraint rules removed the redundant points in overlap region of point cloud and the proposed matching score avoided the mismatch of features. The proposed algorithm can reconstruct 3D scene more quickly and accurately. Furtherly, the reconstructed 3D scene can be utilized to make the path planning and control decision rapidly. Moreover, the proposed algorithm can be utilized to stitch the 3D point cloud of the outdoor scene generated by LIDAR as well as the 3D point cloud of the indoor scene generated by Kinect.

## References

1. Cao, M.: Robust bundle adjustment for large-scale structure from motion. *Multimedia Tools Appl.* **76**(21), 1–25 (2017)
2. Geiger, A., Philip, L., Raquel, U.: Are we ready for autonomous driving? The kitti vision benchmark suite. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2012*, pp. 3354–3361. IEEE (2012)
3. He, Y.: An iterative closest points algorithm for registration of 3D laser scanner point clouds with geometric features. *Sensors* **17**(8), 1862–1872 (2017)
4. Li, Y.: SIFT keypoint removal and injection via convex relaxation. *IEEE Trans. Inf. Forensics Secur.* **11**(8), 1722–1735 (2016)
5. Liang, B., Zheng, L.: Specificity and latent correlation learning for action recognition using synthetic multi-view data from depth maps. *IEEE Trans. Image Process.* **26**(12), 5560–5574 (2017)

6. Mahdi, F.A., Ahmad Fauzi, M.F., Ahmad, N.N.: Image retrieval using most similar highest priority principle based on fusion of colour and texture features. In: Anthony, P., Ishizuka, M., Lukose, D. (eds.) PRICAI 2012. LNCS (LNAI), vol. 7458, pp. 765–770. Springer, Heidelberg (2012). [https://doi.org/10.1007/978-3-642-32695-0\\_70](https://doi.org/10.1007/978-3-642-32695-0_70)
7. Rublee, E.: ORB: an efficient alternative to SIFT or SURF. In: IEEE international conference on Computer Vision (ICCV) 2011, pp. 2564–2571. IEEE (2011)
8. Suhr, J., Ho J.: Noise-resilient road surface and free space estimation using dense stereo. In: Intelligent Vehicles Symposium (IV), pp. 461–466. IEEE (2013)
9. Wang, W., Michael, K.: A variational method for multiple-image blending. *IEEE Trans. Image Process.* **21**(4), 1809–1822 (2012)
10. Wang, X.: An iterative closest point approach for the registration of volumetric human retina image data obtained by optical coherence tomography. *Multimedia Tools Appl.* **76**(5), 6843–6857 (2017)
11. Zheng, S.: A multi-frame graph matching algorithm for low-bandwidth RGB-D SLAM. *Comput. Aided Des.* **7**(8), 107–117 (2016)





# An Improved Particle Filter Target Tracking Algorithm Based on Color Histogram and Convolutional Network

Shasha Gao<sup>(✉)</sup>, Liang Zhou, and Qiang Xie

College of Computer Science and Technology,  
Nanjing University of Aeronautics and Astronautics, Nanjing, China  
gaoshasha@nuaa.edu.cn

**Abstract.** Color feature is mainly adopted in the traditional particle filter method when tracking the target. In view of the problem of failing to track the target caused by background similarity and occlusion, an improved particle filter tracking algorithm based on color histogram and convolution network is proposed, which makes full use of the color feature and convolution feature of the target. Experiments show that compared with the traditional tracking algorithm based on particle filter, the proposed algorithm has a good ability to adapt to changes in the environment around the target.

**Keywords:** Convolutional network · Block color histogram  
Target tracking · Particle filter

## 1 Introduction

With the development of computer vision technology, target tracking has played an important role in life recently, which is mainly used in video surveillance, human-computer interaction, visual navigation and military guidance. For example, in the field of human-computer interaction, the key to computer recognition of human gestures lies in tracking technology. In recent years, researchers have proposed different target tracking algorithms. Nummiaro et al. proposed a particle filter tracking algorithm that combines color features [1]. Ding et al. proposed a particle filter tracking algorithm that combines color features and LBP texture feature, but the problem of target occlusion cannot be well resolved [2]. Dong et al. proposed a meanshift tracking algorithm using color histograms and SIFT features [3]. Tao et al. proposed a color histogram target tracking algorithm with overlapped seed blocks, but when the color of the target is not obvious, which leads to tracking failure [4]. Ji and Wang proposed a target tracking algorithm based on local dynamic sparse representation [5]. Li et al. proposed a target tracking algorithm, which uses the convolution feature and SVM classifier to mark the positive and negative samples of the target, resulting in good tracking results. However, it can cause tracking drift [6]. Wang et al. proposed

a convolutional network tracking algorithm based on acceleration of Gaussian kernel functions, but it could not solve the problem of large area occlusion [7]. Zhang et al. proposed a robust tracking algorithm based on convolutional network without training [8]. Mocuano et al. proposed a tracking algorithm based on offline training for convolutional neural networks using template matching, which requires a large amount of training data samples [9]. However, traditional algorithms cannot accurately track the target when the target is affected by occlusion, illumination and changes in appearance.

This paper proposes a particle filter tracking algorithm that combines color feature and convolution feature. The local and spatial information of the target is fully utilized to represent the state change of the target which leads to robust tracking by integrating color histograms and features extracted from the convolutional network.

## 2 Framework of BCH-CN-PF Algorithm

In this paper, the block color histogram (BCH) and convolutional network (CN) are used to extract the color features and convolution features respectively and the particle filter (PF) framework is used to track the target, which is referred to as an improved particle filter target tracking algorithm based on color histogram and convolutional network (BCH-CN-PF). The framework of the BCH-CN-PF algorithm is shown in Fig. 1. The first step is to initialize. The target image of the first frame is normalized and the color histogram of the target is extracted. Then the next frame is read. The second step is to establish the target model, including the color model, convolution model and the overall observation model.

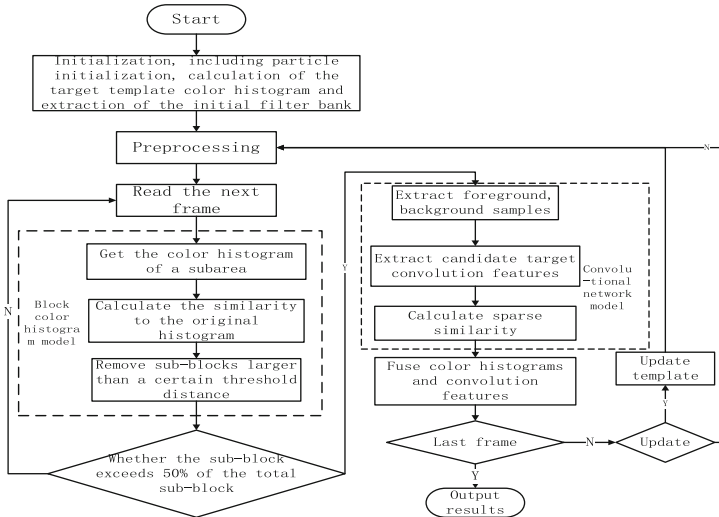


Fig. 1. Framework of BCH-CN-PF algorithm.

Finally, the color feature and convolution feature of the candidate objects are combined. The template is updated by the features learned in the previous frame and the tracking results are output.

### 3 Description of BCH-CN-PF Algorithm

#### 3.1 Image Preprocessing

For the input image  $I$ , it is converted to a image of fixed-size  $n \times n$ , denoted as  $I \in R^{n \times n}$ . A set of local image blocks  $y \in \{Y_1, \dots, Y_l\}, Y_i \in R^{k \times k}$  can be obtained from the original image  $I$  by using the sliding window of size  $k \times k$  to sample at random, where  $Y_i$  is the  $i$ -th local image block. The total number of image blocks is  $l = (n - k + 1) \times (n - k + 1)$ .

#### 3.2 Block Color Histogram

Because the traditional weighted color histogram ignores the color information of the target at the edge, it may lose important information, so a block color histogram method is proposed. First, the target is divided into certain overlapping sub-areas, the method of calculating statistical color model and color space has no difference with the traditional histogram. To reduce computational complexity, the color histogram of the individual sub-blocks is first calculated and then the color histogram covering the sub-regions of the relevant sub-blocks is calculated.

The Bhattacharyya coefficient is used to calculate the similarity between the two models. Assuming that  $p(\mu)$  is candidate target model,  $q$  is target model.

$$\rho(\mu) = \rho(p(\mu), q) = \sum_{u=1}^m \sqrt{p(\mu)q} \tag{1}$$

The degree of similarity is proportional to  $p(\mu)$ . Assume that the distance  $d$  between the two blocks is  $d(\mu) = \sqrt{1 - \rho(\mu)}$ . If  $d$  is larger than the threshold  $d_{thr}$ , the corresponding sub-block is discarded. Count the number of remaining sub-blocks  $N_{rem}$ . If  $N_{rem}/N < 0.5$ , the particle weight is updated to 0. The distance between the target and the candidate target is updated as the average of the distance  $d_{avg}(\mu)$  of the remaining sub-blocks.

Define the observation probability at time  $t$  as:

$$p_1(o_t | s_t^i) = \frac{1}{\sqrt{2\pi}} \exp^{-\eta d_{avg}^2(\mu)/2} \tag{2}$$

Empirical value  $\eta$  is set to 20. The sub-blocks update formula is as follows:

$$q_{ref}' = (1 - \alpha)q_{ref} + \alpha p_{cur} \tag{3}$$

$q_{ref}$  is the color histogram of the sub-block of the reference model,  $p_{cur}$  represents the color histogram of the corresponding sub-block of the current candidate.

### 3.3 Convolutional Network Model

In order to solve the problem of lack of training samples, the convolutional network is used to describe the local feature. The steps are described as follows:

Step 1. Firstly, k-means clustering method is used to select  $d$  image blocks from the  $l$  blocks as the filter template, denoted as  $F^o = \{F_1^o, \dots, F_d^o\} \subset y$ . Given the  $i$ -th filter  $F_i^o \in R^{k \times k}$ , the corresponding feature map of the input image  $I$  is  $S_i^o \in F_i^o \otimes I$ ,  $S_i^o \in R^{(n-k+1) \times (n-k+1)}$  and  $\otimes$  is the convolution operator. The same method is used to obtain a set of templates  $F_i^b = \{F_{i,1}^b, \dots, F_{i,d}^b\} \subset y$  and average background  $F^b = \{F_1^b = (1/m \sum_{i=1}^m F_{i,1}^b), \dots, F_d^b = (1/m \sum_{i=1}^m F_{i,d}^b)\}$ , then convolves with the input image  $I$  to get the feature map  $S_i^b \in F_i^b \otimes I$ . Finally, the feature of target layer is defined as  $S_i = S_i^o - S_i^b = (F_i^o - F_i^b) \otimes I, i = 1, \dots, d$ .

Step 2. A 3-dimensional tensor  $Z \in R^{(n-k+1) \times (n-k+1) \times d}$  is defined to describe the deep complex features by superimposing the  $d$  convolution kernels and we use a sparse representation to approximate  $vec(Z)$ .

$$\hat{z} = \arg \min_z \frac{1}{2} \|z - vec(Z)\|_2^2 + \lambda \|z\|_1 \quad (4)$$

Then, we use the soft threshold method to get the solution, where  $sign(\cdot)$  is a symbolic function.

$$\hat{z} = sign(vec(Z)) \max(0, |vec(Z)| - \lambda) \quad (5)$$

Step 3. The low-pass filtering method is used to update model:

$$z_t = (1 - \rho)z_{t-1} + \rho \hat{z}_{t-1} \quad (6)$$

where  $\rho$  is the filter coefficient,  $z_t$  is the target template at frame  $t$  and  $\hat{z}_{t-1}$  is the sparse expression of  $z_{t-1}$ .

The observation model is defined by formula (7):

$$p_2(o_t | s_t^i) = \exp^{-\|z_t - z_t^i\|_2^1} \quad (7)$$

where  $z_t^i = vec(Z_t^i) \odot \beta$ ,  $z_t^i$  is the representation of the  $i$ -th candidate sample at frame  $t$ ,  $\odot$  represents the product of the elements, if  $z_t(i)$  is 0,  $\beta$  is 0, else is 1.

### 3.4 Improved Particle Filtering Algorithm

Assume that the target parameters are independent and in order to improve the efficiency of computation, we define a motion model:

$$p(s_t | s_{t-1}) = \lambda_t N(s_{t-1}, \sigma^2) \quad (8)$$

where  $\sigma^2$  is the variance of the target motion state. If  $N_{rem}/N < 0.5$ ,  $\lambda_t = 0$ ; else  $\lambda_t = 1$ . Then particles similar to the target can be filtered out in the block color histogram processing flow.

Combining target color feature and convolution feature, the observed probability density function for each particle is

$$p(o_t|s_t^i) = \varepsilon p_1(o_t|s_t^i) + (1 - \varepsilon)p_2(o_t|s_t^i) \tag{9}$$

The parameter  $\varepsilon$  is used to adjust the proportion of global features and local features in the total observed probability, which is set to  $\varepsilon < 0.5, 0 \leq \varepsilon \leq 1$ .

Substitute (2), (7) to usual weight formula  $w_t^i = p(o_t|s_t^i) \cdot w_{t-1}^i$ , we obtain the posterior estimate of the current moment and output it, which is expressed as

$$\hat{s}_t = \sum_{i=1}^N s_t^i w_t^i = \sum_{i=1}^N s_t^i w_{t-1}^i \left( \varepsilon \frac{1}{\sqrt{2\pi}} \exp^{-\eta d_{avg}^2(y)/2} + (1 - \varepsilon) \exp^{-\|z_t - z_t^i\|_2^2} \right) \tag{10}$$

### 4 Experiments and Analysis

The experiment is based on the MATLAB2016b platform using the test sequence of tracker\_benchmark 1.0 [10], which provides a benchmark for different algorithms. The input image size is set to  $32 \times 32$  and the sliding window size to  $6 \times 6$ . Then the number of filters is set to  $d = 100$ , the filter coefficient  $\rho$  is set to 0.95 and the number of particles is set to 350,  $d_{thr}$  is set to 0.25 and the particle observation probability is set to 0.2. The value of  $\alpha$  is 0.5 and  $\varepsilon$  is 0.35. All the parameters are fixed in the experiment. The algorithm and other three tracking algorithms CPF (Color-Based Probabilistic Tracking) [11], VTS (Tracking by Sampling Trackers) [12] and VTD (Visual Tracking Decomposition) [13] are compared. The comparison algorithms all use particle filter as tracking framework and the difference lies in that the methods of extracting features.

#### 4.1 Quantitative Analysis

The precision plot and success plot of the CH-CN-PF algorithm and the comparison algorithm are given below.

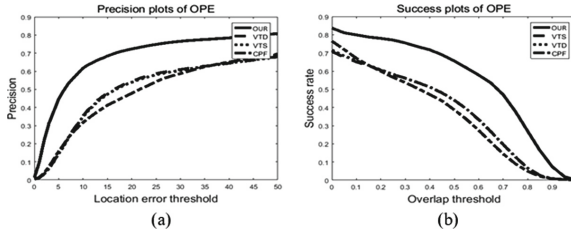


Fig. 2. The precision plot and success plot.

It can be seen from Fig. 2 that the CH-CN-PF algorithm leads in the overall performance. This is due to the use of convolution feature and color feature in

the representation of the target feature. When the target size changes, the target image is normalized at first and the extracted local convolution features still maintain the geometric characteristics of the target, so the algorithm is robust to this case. When the target is affected by occlusion, the block color histogram can remove the occluded local color histogram and reconstruct the candidate target color feature. In summary, the CH-CN-PF algorithm has outstanding performance in comprehensive performance and high tracking accuracy.

## 4.2 Qualitative Analysis

- (1) Changes in illumination and target posture affect the tracking effect. As time of the tracker drift increases, the cumulative error increases and the tracker will lose the target. The color feature of the image is not obvious but the convolutional net-work can extract its depth characteristics, so the CH-CN-PF algorithm can track the target (Fig. 3).
- (2) The occlusion of street lights and tree trunks in the sequence of images has effect on tracking. The proposed model has certain robustness to occlusion, while other algorithms cannot accurately track the target (Fig. 4).



Fig. 3. Tracking results of 4 algorithms under illumination (Shaking image sequence).



Fig. 4. Tracking results of 4 algorithms under occlusion (David3 image sequence).

## 5 Summary

An improved CH-CN-PF algorithm is proposed based on the particle filter tracking algorithm, which merges the color feature of the target and the feature extracted by the convolutional network to construct the observation model. Compared with the traditional tracking algorithm based on particle filter, the tracking success rate and accuracy rate have improved significantly. Experimental results show that the CH-CN-PF algorithm can improve the adaptability to the environment.

## References

1. Nummiaro, K., Koller-Meier, E., Gool, L.V.: An adaptive color-based particle filter. *Image Vis. Comput.* **21**(1), 99–110 (2003)
2. Ding, D., Jiang, Z., Liu, C.: Object tracking algorithm based on particle filter with color and texture feature. In: *Proceedings of Control Conference*, pp. 4031–4036. IEEE (2016)
3. Dong, W., Chang, F., Li, T.: Adaptive block target tracking method based on color histogram and SIFT features. *J. Electron. Inf. Technol.* **35**(4), 770–776 (2013)
4. Tao, L.: Object tracking based on sub-block color histogram and particle filter. *Comput. Eng. Appl.* **48**(7), 165–168 (2012)
5. Ji, Z., Wang, W.: Object tracking based on local dynamic sparse model. *J. Vis. Commun. Image. Represent.* **28**, 44–52 (2015)
6. Li, J., Zhou, X., Chan, S., et al.: Object tracking using a convolutional network and a structured output SVM. *Comput. Vis. Media* **4**, 1–11 (2017)
7. Wang, H., Liu, P., Luo, Y., et al.: Convolutional neural network tracking algorithm based on gauss kernel function. In: *CAAI Transactions on Intelligent Systems* (2016)
8. Zhang, K., Liu, Q., Wu, Y., et al.: Robust visual tracking via convolutional networks without training. *IEEE Trans. Image Process. Publ. IEEE Sig. Process. Soc.* **25**(4), 1779–1792 (2016)
9. Mocanu, B., Tapu, R., Zaharia, T.: Object tracking using deep convolutional neural networks and visual appearance models. In: Blanc-Talon, J., Penne, R., Philips, W., Popescu, D., Scheunders, P. (eds.) *ACIVS 2017. LNCS*, vol. 10617, pp. 114–125. Springer, Cham (2017). [https://doi.org/10.1007/978-3-319-70353-4\\_10](https://doi.org/10.1007/978-3-319-70353-4_10)
10. Wu, Y., Lim, J., Yang, M.H.: Online object tracking: a benchmark. In: *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2411–2418. IEEE Computer Society (2013)
11. Pérez, P., Hue, C., Vermaak, J., Gangnet, M.: Color-based probabilistic tracking. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) *ECCV 2002. LNCS*, vol. 2350, pp. 661–675. Springer, Heidelberg (2002). [https://doi.org/10.1007/3-540-47969-4\\_44](https://doi.org/10.1007/3-540-47969-4_44)
12. Kwon, J., Lee, K.M.: Tracking by sampling trackers. In: *IEEE International Conference on Computer Vision*, pp. 1195–1202. IEEE (2011)
13. Kwon, J., Lee, K.M.: Visual tracking decomposition. In: *Computer Vision and Pattern Recognition*, pp. 1269–1276. IEEE (2010)



# Mini-Batch Variational Inference for Time-Aware Topic Modeling

Tomonari Masada<sup>1</sup>✉ and Atsuhiro Takasu<sup>2</sup>

<sup>1</sup> Nagasaki University, 1-14 Bunkyo-machi, Nagasaki-shi, Nagasaki, Japan  
masada@nagasaki-u.ac.jp

<sup>2</sup> National Institute of Informatics, 2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo, Japan  
takasu@nii.ac.jp

**Abstract.** This paper proposes a time-aware topic model and its mini-batch variational inference for exploring chronological trends in document contents. Our contribution is twofold. First, to extract topics in a time-aware manner, our method uses two vector embeddings: the embedding of latent topics and that of document timestamps. By combining these two embeddings and applying the softmax function, we have as many word probability distributions as document timestamps for each topic. This modeling enables us to extract remarkable topical trends. Second, to achieve memory efficiency, the variational inference is implemented as mini-batch gradient ascent maximizing the evidence lower bound. This enables us to perform parameter estimation in the way similar to neural networks. Our method was actually implemented with deep learning framework. The evaluation results show that we could improve test set perplexity by using document timestamps and also that our test perplexity was comparable with that of collapsed Gibbs sampling, which is less efficient in memory usage than the proposed inference.

**Keywords:** Topic modeling · Variational inference  
Time-aware analysis

## 1 Introduction

This paper proposes a topic model using document timestamps, which is an extension of latent Dirichlet allocation (LDA) [3], and provides an inference for the proposed model. Our contribution is twofold. First, we use two vector embeddings in our model: the vector embedding of latent topics and that of document timestamps. While more sophisticated methods have been proposed for using timestamps in topic modeling [2, 4], vector embedding is an inexpensive way to incorporate covariates like timestamps. Second, we provide a mini-batch variational inference for the proposed model. An important merit of mini-batch inference is memory efficiency, because there is no need to keep any corpus-wide information. These two contributions, explained in the rest of this chapter, provide a viewpoint from which we can regard our time-aware topic modeling as



sharing common features with neural network-based data modeling. We actually implemented the proposed inference by using PyTorch.<sup>1</sup>

The first contribution concerns modeling. Topic models use two types of categorical distributions: per-document categorical distributions over topics and per-topic categorical distributions over words. Each document is modeled with a categorical distribution defined over topics. We denote the number of topics by  $K$ . Let  $\theta_{d,k}$  refer to the probability of the topic  $k$  in the document  $d$ . We introduce no modification with respect to  $\theta_{d,k}$  and estimate the corresponding posterior parameters as in [3]. Each topic is in turn represented as a categorical distribution defined over vocabulary words. We denote the vocabulary size by  $V$  and refer to the probability of the word  $v$  in the topic  $k$  by  $\phi_{k,v}$ . The proposed model obtains  $\phi_{k,v}$  by applying the softmax function to the  $V$ -dimensional vector embedding  $\mathbf{w}_k$  of the topic  $k$  plus bias, i.e.,  $\phi_{k,v} = \frac{\exp(w_{k,v} + b_v)}{\sum_{v'} \exp(w_{k,v'} + b_{v'})}$ , where  $b_v$  is shared by all topics. Let  $\mathbf{W}$  be the  $V \times K$  matrix whose  $k$ -th column is  $\mathbf{w}_k$ . Then  $\boldsymbol{\phi}_k = (\phi_{k,1}, \dots, \phi_{k,V})^\top$  is written as  $\boldsymbol{\phi}_k = \text{Softmax}(\mathbf{W}\mathbf{e}_k + \mathbf{b})$ , where  $\mathbf{e}_k$  is the one-hot vector whose  $k$ -th element is 1 and all other elements are 0. While this formulation is similar to that in [7], we here provide a viewpoint from which  $\boldsymbol{\phi}_k$  can be regarded as the softmax output of a single-layer feed-forward network. We adopt the same approach also for document timestamps. Assume that there are  $T$  different timestamps and that  $\mathbf{u}_t$  is the  $V$ -dimensional vector embedding of the timestamp  $t$ . We then obtain a *time-aware* version of  $\boldsymbol{\phi}_k$  as  $\boldsymbol{\phi}_{k,t} = \text{Softmax}(\mathbf{W}\mathbf{e}_k + \mathbf{U}\mathbf{o}_t + \mathbf{b})$ , where  $\mathbf{U}$  is the  $V \times T$  matrix whose  $t$ -th column is  $\mathbf{u}_t$ , and  $\mathbf{o}_t$  is the one hot vector whose  $t$ -th element is 1.  $\boldsymbol{\phi}_{k,t} = (\phi_{k,t,1}, \dots, \phi_{k,t,V})^\top$  represents the word probabilities in the topic  $k$  at each time point (cf. Fig. 1). Our evaluation experiment showed that the time-aware modeling led to an improvement in several cases.



**Fig. 1.** Word cloud presentation of the same topic at three different time points. This topic, seemingly related to HCI, extracted from 827,141 paper titles in DBLP data set (cf. Sect. 3). The number of topics was set to 50. The font size is determined by  $\phi_{k,t,v}$ , the probability of the word  $v$  in the topic  $k$  at the time point  $t$ . The left, center, and right panels correspond to the time points 1998, 2005, and 2012, respectively.

The second contribution concerns inference. Mini-batch based variational inferences have already been proposed for topic models [4, 9]. However, these

<sup>1</sup> <https://github.com/pytorch>.

inferences require the knowledge of the total number of training documents. In this paper, we provide a mini-batch learning for the proposed model, where we perform the parameter estimation in a similar manner to that for neural networks. We iterate over mini-batches and conduct gradient ascent where the evidence lower bound (ELBO) is maximized. While we need to choose an appropriate optimization algorithm like Adam [11] and to adjust its learning rate, this is what we usually do for training neural networks. The experimental results show that the test perplexity achieved by the proposed inference was comparable to that achieved by collapsed Gibbs sampling (CGS) for LDA [8]. While CGS often gives better perplexity than variational inference [1], it is less efficient in memory use. The proposed method works even when the number of documents is prohibitively large for CGS.

## 2 Method

While there are many ways to perform inference for topic models, we adopt variational inference [3,4,9], because it allows us to perform inference as optimization. Our model is a modification of the vanilla LDA [3]. A symmetric Dirichlet prior  $\text{Dirichlet}(\alpha)$  is assigned to the per-document categorical distributions over topics as in the vanilla LDA. In contrast, no prior is assigned to the per-topic categorical distributions over words, because the word probabilities are estimated differently from the vanilla LDA. The ELBO of LDA is given as

$$\begin{aligned} \sum_d \log p(\mathbf{x}_d; \alpha, \Phi) &\geq \sum_{d,v,k} n_{d,v} \gamma_{d,v,k} \log \phi_{k,v} \\ &+ \sum_{d,k} \left( \alpha + \sum_v n_{d,v} \gamma_{d,v,k} - \lambda_{d,k} \right) \left\{ \Psi(\lambda_{d,k}) - \Psi \left( \sum_{k'} \lambda_{d,k'} \right) \right\} \\ &- \sum_{d,v,k} n_{d,v} \gamma_{d,v,k} \log \gamma_{d,v,k} + \log \Gamma(K\alpha) - K \log \Gamma(\alpha) \end{aligned} \quad (1)$$

where  $\mathbf{x}_d$  is the bag-of-words representation of the document  $d$ ,  $n_{d,v}$  is the frequency of the word  $v$  in  $d$ ,  $\gamma_{d,v,k}$  is the responsibility of the word  $v$  with respect to the topic  $k$  in  $d$ , and  $\lambda_{d,k}$  is the posterior parameter for the topic  $k$  in  $d$ .  $\Psi$  is the digamma function. The document-specific parameters  $\lambda_{d,k}$  and  $\gamma_{d,v,k}$  are estimated as in [3,4,9], and we regard  $\alpha$  as free parameter. However, we estimate  $\Phi = \{\phi_1, \dots, \phi_K\}$ , the parameter vectors of the per-topic categorical distributions over words, differently. In our model,  $\phi_k$  is given by  $\phi_k = \text{Softmax}(\mathbf{W}e_k + \mathbf{b})$  as discussed in Sect. 1, where the  $k$ -th column of  $\mathbf{W}$  can be viewed as a vector embedding of the topic  $k$ . While we use no prior, the bias term  $b_v$  for each  $v$  plays a similar role to Dirichlet smoothing [1,5].

Our proposal also considers the usage of covariates like timestamps, authors, venues, etc. In this paper, we explore chronological trends in document contents and thus make the per-topic word probabilities *time-aware* as  $\phi_{k,t,v} = \frac{\exp(w_{k,v} + u_{t,v} + b_v)}{\sum_{v'} \exp(w_{k,v'} + u_{t,v'} + b_{v'})}$ .  $\mathbf{U} = (u_{t,v})^\top$  is a weight matrix modeling the dependency of word probabilities on timestamps. Each column of  $\mathbf{U}$  can be regarded

**Algorithm 1.** Mini-batch variational inference

---

```

1: Input:  $n_{d,v}$  for each mini-batch,  $\alpha$ , and  $\eta$ 
2: Initialize  $\mathbf{W}$ ,  $\mathbf{U}$ , and  $\mathbf{b}$ 
3: while True do
4:    $\phi_{k,t} \leftarrow \text{Softmax}(\mathbf{W}e_k + \mathbf{U}o_t + \mathbf{b})$ 
5:   Get next mini-batch
6:   for each document  $d$  in mini-batch do
7:      $t \leftarrow$  timestamp of document  $d$ 
8:     Initialize  $\gamma_{d,v,k}$  randomly
9:     repeat
10:       $\lambda_{d,k} \leftarrow \alpha + \sum_v n_{d,v} \gamma_{d,v,k}$ 
11:       $\gamma_{d,v,k} \leftarrow \propto \exp(\Psi(\lambda_{d,k})) \times \phi_{k,t,v}$ 
12:    until change in  $\lambda_{d,k}$  is negligible
13:   end for
14:   Make computational graph of the negative of ELBO and backpropagate
15:   Update  $\mathbf{W}$ ,  $\mathbf{U}$ , and  $\mathbf{b}$  by using  $\eta$ 
16:   Update  $\eta$  if necessary
17: end while

```

---

as the vector embedding of the timestamp  $t$ . As discussed in Sect. 1,  $\phi_{k,t}$  can be written as  $\phi_{k,t} = \text{Softmax}(\mathbf{W}e_k + \mathbf{U}o_t + \mathbf{b})$ . We estimate  $\mathbf{W}$ ,  $\mathbf{U}$ , and  $\mathbf{b}$  by maximizing the ELBO in Eq. (1).

The pseudo-code is given in Algorithm 1. The inputs are the frequency  $n_{d,v}$  of the word  $v$  in the document  $d$ , the Dirichlet prior parameter  $\alpha$ , and the initial learning rate  $\eta$ . Thanks to the broadcasting mechanism in PyTorch, we can write the word probabilities as `torch.nn.functional.softmax(W.unsqueeze(2) + U.unsqueeze(1) + b.unsqueeze(1).unsqueeze(2), dim=0)`. In the evaluation experiment, we sometimes obtained better results with DropConnect [15], which was implemented by applying dropout to  $\mathbf{W}$  and  $\mathbf{U}$ . The extent of the improvement reached a significant level.  $\mathbf{W}$  and  $\mathbf{U}$  were initialized by the Gaussian distribution with zero-mean and a variance of 0.01.  $\mathbf{b} = (b_1, \dots, b_V)$  were initialized to zero. We used Adam optimizer [11] with the learning rate update  $\eta(1 + m/500)^{-0.7}$  for the  $m$ -th mini-batch, where  $\eta$  was the initial learning rate. Further,  $\eta$  was halved every 500 mini-batches until 2,000 mini-batches were seen.

We give an additional discussion on modeling. Our method combines the parameters indexed by the tuple  $(k, v)$  with those indexed by  $(t, v)$  to make word probabilities dependent on timestamps. However, we could also introduce the parameters  $r_{k,t,v}$  indexed explicitly by the 3-tuple  $(k, t, v)$  as  $\phi_{k,t,v} = \frac{\exp(w_{k,v} + u_{t,v} + r_{k,t,v} + b_v)}{\sum_{v'} \exp(w_{k,v'} + u_{t,v'} + r_{k,t,v'} + b_{v'})}$  [7, 13]. This approach can separate out the words not depending on topics but exclusively depending on timestamps (e.g. ‘2001’, ‘December’, etc.) from the words depending both on topics and on timestamps. While both our approach and this have a time complexity of  $O(KTV)$  per mini-batch, the latter consumes memory space for  $K \times T \times V$  variables each requiring gradient. Since we could obtain interesting time-dependencies as presented in Fig. 1, we did not introduce the parameters explicitly indexed by  $(k, t, v)$ .

### 3 Experiment

In the evaluation experiment, we used the four document sets given in Table 1, where the total number of documents, the vocabulary size, and the number of different document timestamps are denoted by  $D$ ,  $V$ , and  $T$ , respectively. ‘MAI’ is a set of newswire articles from the Mainichi, a Japanese newspaper. A morphological analysis was performed by using MeCab.<sup>2</sup> The dates of the articles range from November 1, 2007 to May 15, 2008. We split the date range into 13 slices of nearly equal size and used the slices as document timestamps. ‘TDT4’ is a set of English documents from TDT4 Multilingual Text and Annotations.<sup>3</sup> The dates of the documents range from December 1, 2000 to January 31, 2001. We split the date range into 15 slices of nearly equal size. ‘DBLP’ is a set of paper titles from DBLP web site.<sup>4</sup> We regarded each title as a single document. The publication years ranging from 1998 to 2012 were used as document timestamps. ‘STOV’ is a subset of the questions in the StackOverflow data set available at Kaggle.<sup>5</sup> We used as document timestamps the months on which the questions were published, ranging from January 2014 to December 2016. For all data sets, English words were lower-cased, and highly frequent words and infrequent words were removed. The training:validation:test ratio was 8:1:1 for all data sets. The mini-batch size was 200 for MAI, TDT4, and STOV and was 10,000 for DBLP, because DBLP was a set of very short documents.

**Table 1.** Specifications of data sets

	$D$	$V$	$T$		$D$	$V$	$T$
MAI	32,775	15,161	13	DBLP	1,034,067	18,940	15
TDT4	96,246	15,153	15	STOV	25,608	13,184	34

We compared the following three approaches for topic extraction: the mini-batch inference for the topic model with document timestamps, the mini-batch inference for the topic model without document timestamps, and collapsed Gibbs sampling (CGS) [8] for the vanilla LDA. These compared methods are referred to by ‘VB w/ t’, ‘VB’, and ‘CGS’, respectively in Table 2, which summarizes the evaluation results. For the number of latent topics, we tested the following two settings:  $K = 50$  and  $K = 100$ . The comparison was performed in terms of test perplexity [3]. We tuned free parameters based on the perplexity computed over validation set. For ‘VB w/ t’ and ‘VB’, the initial learning rate  $\eta$  of Adam and the parameter  $\alpha$  of the symmetric Dirichlet prior were tuned based on validation perplexity. The tuned values of  $\eta$  and  $\alpha$  are given in Table 2. With respect to

<sup>2</sup> <http://taku910.github.io/mecab/>.

<sup>3</sup> <https://catalog.ldc.upenn.edu/LDC2005T16>.

<sup>4</sup> <http://dblp.uni-trier.de/xml/>.

<sup>5</sup> <https://www.kaggle.com/stackoverflow/rquestions>.

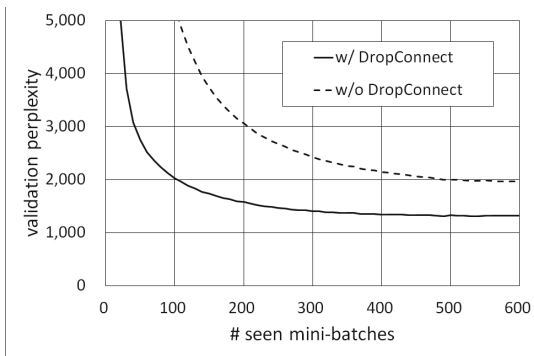
‘CGS’,  $\alpha$  and the parameter  $\beta$  of the symmetric Dirichlet prior assigned to the per-topic word categorical distributions were tuned based on validation perplexity [1]. The tuned values of  $\alpha$  and  $\beta$  are given in Table 2. The validation and test perplexities were computed by using fold-in procedure [1], where randomly selected two-thirds of word tokens were used for obtaining topic posterior probabilities in each document. We repeated the test perplexity computation ten times each over a different set of randomly chosen two-thirds and then calculated the mean and standard deviation of the ten perplexities.

**Table 2.** Evaluation results in terms of test set perplexity

	$K$	Method	Test set perplexity (mean $\pm$ stdev)	Free parameters ( $\eta$ : learning rate) ( $\alpha, \beta$ : Dirichlet)
MAI	100	VB w/ t	<b>1115.64 <math>\pm</math> 3.94</b>	$\eta = .07, \alpha = .03$
		VB	1138.99 $\pm$ 3.11	$\eta = .07, \alpha = .03$
		CGS	1130.93 $\pm$ 3.04	$\alpha = .02, \beta = .05$
	50	VB w/ t	<b>1283.05 <math>\pm</math> 5.74</b>	$\eta = .08, \alpha = .02$
		VB	1315.10 $\pm$ 5.94	$\eta = .08, \alpha = .02$
		CGS	1336.75 $\pm$ 5.56	$\alpha = .04, \beta = .15$
TDT4	100	VB w/ t	1480.82 $\pm$ 3.85	$\eta = .06, \alpha = .02$
		VB	1481.13 $\pm$ 3.79	$\eta = .06, \alpha = .04$
		CGS	<b>1425.64 <math>\pm</math> 3.08</b>	$\alpha = .02, \beta = .10$
	50	VB w/ t	1587.45 $\pm$ 3.25	$\eta = .05, \alpha = .01$
		VB	1589.99 $\pm$ 3.10	$\eta = .10, \alpha = .02$
		CGS	1586.48 $\pm$ 3.05	$\alpha = .05, \beta = .05$
DBLP	100	VB w/ t	<b>1274.45 <math>\pm</math> 3.88</b>	$\eta = 6e-4, \alpha = .09^\dagger$
		VB	1325.30 $\pm$ 5.25	$\eta = 8e-4, \alpha = .08^\dagger$
		CGS	1341.07 $\pm$ 4.84	$\alpha = .03, \beta = .04$
	50	VB w/ t	<b>1300.09 <math>\pm</math> 5.69</b>	$\eta = 5e-4, \alpha = .12^\dagger$
		VB	1335.71 $\pm$ 5.72	$\eta = 6e-4, \alpha = .10^\dagger$
		CGS	1384.36 $\pm$ 6.21	$\alpha = .04, \beta = .05$
STOV	100	VB w/ t	1083.10 $\pm$ 4.82	$\eta = .015, \alpha = .20^\dagger$
		VB	1125.78 $\pm$ 5.62	$\eta = .010, \alpha = .30^\dagger$
		CGS	<b>1042.18 <math>\pm</math> 4.19</b>	$\alpha = .05, \beta = .01$
	50	VB w/ t	1255.72 $\pm$ 3.59	$\eta = .018, \alpha = .30$
		VB	1266.16 $\pm$ 3.43	$\eta = .020, \alpha = .35$
		CGS	<b>1178.20 <math>\pm</math> 6.16</b>	$\alpha = .07, \beta = .04$

As Table 2 shows, ‘VB w/ t’ gave the best result among the three compared methods when  $K = 50$  or  $100$  for MAI and when  $K = 50$  or  $100$  for DBLP. Even when we adopt the topic model using no timestamps, the mini-batch inference

gave the test perplexity better than that of CGS when  $K = 50$  for MAI and when  $K = 50$  or 100 for DBLP. Since CGS often gives better test perplexity than variational inference [1], it can be said that the proposed mini-batch inference is a promising alternative to CGS for data sets prohibitively large for CGS even when we use no timestamps. Another important observation is that DropConnect [15] led to a remarkable improvement when  $K = 50$  or 100 for DBLP and when  $K = 100$  for STOV, i.e., the cases marked by dagger ‘†’ in Table 2. DropConnect worked for both ‘VB w/ t’ and ‘VB’. Figure 2 shows to what extent DropConnect could improve the validation perplexity for DBLP data set when  $K = 50$ . The horizontal axis gives the number of seen mini-batches. The vertical axis gives the validation perplexity during the course of inference. DropConnect could improve the validation perplexity by around 500 for this case.



**Fig. 2.** Comparing the inference with DropConnect (solid line) to that without it (dashed line) in terms of validation perplexity for DBLP data set when  $K = 50$ . The horizontal axis gives the number of seen mini-batches. The validation perplexity given by the inference without DropConnect (dashed line) reached a stable value, around 1,870, after 2,000 mini-batches were seen. The validation perplexity given by the inference with DropConnect reached around 1,320.

In Fig. 1, we present a latent topic, extracted by the proposed method from DBLP data set, as word clouds.<sup>6</sup> In the three panels we provide top 50 words according to their probabilities  $\phi_{k,t,v}$  at three different time points, i.e., 1998, 2005, and 2012, respectively. In this topic, seemingly related to HCI, the word ‘object’ is more relevant in 1998 than in 2005 and 2012. In contrast, the word ‘application’ is more relevant in 2005 than in 1998 and 2012, and the words ‘interaction’ and ‘augmented’ are more relevant in 2012 than in 1998 and 2005. In this manner, our approach can explore chronological trends in document contents.

<sup>6</sup> [https://github.com/amueller/word\\_cloud](https://github.com/amueller/word_cloud).

## 4 Related Work

There have already been proposals for combining topic modeling with neural networks. Mikolov and Zweig [12] devised an RNN-based language model, where LDA is used for obtaining additional context information in language modeling. However, this is neither a new topic model nor a new inference for topic models. Srivastava and Sutton [14] proposed ProdLDA, where the variational parameters are computed by using encoder network in VAE [10]. Dieng et al. [6] proposed TopicRNN, where RNN is used to obtain per-document word probabilities. Both proposals use the softmax function to carry out the word-level mixture over per-topic word probabilities after collapsing  $z_{d,i}$ , i.e., latent variables each representing the topic assignment of the  $i$ -th word token in the document  $d$ . However, the collapsing of  $z_{d,i}$  makes both models widely deviate from the vanilla LDA [3]. Our method has tried to keep modification to a minimum in making per-topic word probabilities of the vanilla LDA time-aware.

## 5 Conclusion

This paper proposed a topic model using document timestamps as covariate and provided a mini-batch inference implemented by using a deep learning framework. Our mini-batch variational inference updates per-topic word probabilities in a similar manner to neural network parameters. We also showed that DropConnect worked. It is an important future research direction to consider covariates other than timestamps, e.g. authors, affiliations of authors, publication venues, etc., in the way similar to [13] for exploring document contents in a wider societal context.

## References

1. Asuncion, A., Welling, M., Smyth, P., Teh, Y.W.: On smoothing and inference for topic models. In: Proceedings of UAI, pp. 27–34 (2009)
2. Blei, D.M., Lafferty, J.D.: Dynamic topic models. In: Proceedings of ICML, pp. 113–120 (2006)
3. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. *J. Mach. Learn. Res.* **3**, 993–1022 (2003)
4. Broderick, T., Boyd, N., Wibisono, A., Wilson, A.C., Jordan, M.I.: Streaming variational Bayes. In: Proceedings of NIPS, pp. 1727–1735 (2013)
5. Chen, S.F., Goodman, J.: An empirical study of smoothing techniques for language modeling. In: Proceedings of ACL, pp. 310–318 (1996)
6. Dieng, A.B., Wang, C., Gao, J., Paisley, J.: TopicRNN: a recurrent neural network with long-range semantic dependency. [arXiv:1611.01702](https://arxiv.org/abs/1611.01702) (2016)
7. Eisenstein, J., Ahmed, A., Xing, E.P.: Sparse additive generative models of text. In: Proceedings of ICML, pp. 1041–1048 (2011)
8. Griffiths, T.L., Steyvers, M.: Finding scientific topics. *Proc. Natl. Acad. Sci. U.S.A.* **101**(Suppl. 1), 5235–5288 (2004)

9. Hoffman, M.D., Blei, D.M., Bach, F.: Online learning for latent Dirichlet allocation. In: Proceedings of NIPS, pp. 856–864 (2010)
10. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. In: Proceedings of ICLR (2014)
11. Kingma, D.P., Ba, J.L.: Adam: a method for stochastic optimization. In: Proceedings of ICLR (2015)
12. Mikolov, T., Zweig, G.: Context dependent recurrent neural network language model. In: Proceedings of IEEE Spoken Language Technology, Workshop, pp. 234–239 (2012)
13. Roberts, M.E., Stewart, B.M., Tingley, D., Airoidi, E.M.: The structural topic model and applied social science. In: Proceedings of Advances in Neural Information Processing Systems Workshop on Topic Models: Computation, Application, and Evaluation (2013)
14. Srivastava, A., Sutton, C.: Autoencoding variational inference for topic models. In: Proceedings of ICLR (2017)
15. Wan, L., Zeiler, M., Zhang, S., Cun, Y.L., Fergus, R.: Regularization of neural networks using DropConnect. In: Proceedings of ICML, pp. 1058–1066 (2013)





# Using Differential Evolution to Estimate Labeler Quality for Crowdsourcing

Chen Qiu<sup>1</sup>, Liangxiao Jiang<sup>1,2</sup>(✉), and Zhihua Cai<sup>1,2</sup>

<sup>1</sup> Department of Computer Science,  
China University of Geosciences, Wuhan, China  
ljjiang@cug.edu.cn

<sup>2</sup> Hubei Key Laboratory of Intelligent Geo-Information Processing,  
China University of Geosciences, Wuhan, China

**Abstract.** Crowdsourcing has emerged as an effective paradigm for accomplishing various intelligent tasks at low costs. However, the labels provided by non-expert crowdsourcing labelers often appear various quality as labelers possess wide-ranging levels of competence. This raises the significant challenges of estimating the true answers for tasks and the reliability of the labelers. Of numerous approaches to estimating labeler quality, expectation-maximization (EM) is widely used by maximizing the likelihood estimates of labeler quality from the observed multiple labels. However, EM-based approaches are easily trapped into local optima. In this paper we use a weight vector to represent the quality (reliability) of corresponding labelers and then using differential evolution (DE) to search optimal weights for different labelers. The experimental results validate the effectiveness of the proposed approach.

**Keywords:** Crowdsourcing · Labeler quality · Differential evolution

## 1 Introduction

In supervised learning, a training instance always consists of a  $d$ -dimensional feature vector and known label(s). It is expensive and time-consuming to acquire the known label(s) from domain experts for supervised learning in many domains. Crowdsourcing [5] is becoming an appealing methodology for collecting labeled data for machine learning, which is demonstrated by Amazon mechanical Turk, Netflix, and ESP game. Unfortunately, the label quality collected from online crowdsourcing service is often unsatisfying: single non-expert labeler may provide incorrect response which can affect the learning algorithms. It may be caused by personal preference, low payment for each task, and varying cognitive abilities. Repeated labeling essentially refers to crowd labelers who provide multiple labels for each instance, which improves the quality of labels. Majority Voting is the simplest and the most straightforward method for integrating multiple

---

This work was partially supported by NSFC (U1711267, 61773355).

labels, before feeding the data to supervised learning algorithm. Nevertheless, this method assumes that all labelers have the same reliability. It is obvious that the assumption of the same reliability is rarely true in real-world application, which would harm its performance in the application of complex labeling tasks.

By relaxing the unrealistic assumption, more sophisticated methods for jointly estimating true answers and labeler quality have been proposed to improve the quality of crowd data. These methods refine probabilistic models of labeling process and show better performance than majority voting. The basic idea of these probabilistic models is to use the Expectation-Maximization (EM) to make quality estimation. Relevant EM strategies are used to iteratively estimate the true label and labeler quality [2–4, 7, 10]. EM has also been used to separate varying labelers into different groups. Although these EM-based algorithms have given promising results, there are challenges in achieving robust performance when considering diverse data sets due to the inherent defects of EM [13]. A further limitation of EM is that it is easily trapped into local optima. It may lead to poor results. This can be addressed through a differential evolution-based method, which is the focus of this paper.

Differential evolution (DE) is an efficient population-based stochastic search technique for solving optimization problems over continuous space. DE algorithm aims at evolving a population of parameter vectors towards the global optimum. Note that learning or optimisation typically refers to the strategy of finding the best set of parameters that best describes the training data. In crowdsourcing, the estimation assignment refers to the probability distribution of labelers’ quality and true label for each task, which can be explored via DE. Differential evolution has been applied to pattern classification and clustering [1, 6]. But it is rarely applied to crowdsourcing. We believe jointly estimation of labeler quality and true labels in crowdsourcing tasks can be addressed through DE.

In this paper, we present a differential evolution-based algorithm to estimate the quality of labelers. The method searches optimal weights for different labelers through a variety of labeler weighted strategies. Experiments demonstrated that the proposed differential evolution-based weighted method for crowdsourcing (DEW) can successfully estimate the reliability of labelers and hidden labels for differential learning tasks, and its performance outperforms other label integration methods.

The rest of the paper is organised as follows. Section 2 proposes our labeler weighted method based on differential evolution (DEW). Section 3 describes experimental datasets and results. Section 4 concludes the paper.

## 2 A Differential Evolution-Based Weighted Consensus Method in Crowdsourcing

For a crowdsourcing system, we collect labels for  $n$  tasks from  $J$  labelers. Each task can be described as  $x_i \in R^d, i = 1, \dots, n$  recorded in the matrix  $X$ . The crowd labels are collected in the matrix  $L$  so that  $l_{i,j} \in \{-1, 0, +1\}$  denotes the label of task  $i$  given by labeler  $j$ . The special value -1 denotes that the task is

not labeled. Thus, the training data can be noted as  $T = (X, Y, L)$ . The true label  $Y$  is unknown. Let  $U$  be the set of labelers in an crowdsourcing system. We account for each labeler's quality by assigning a parameter  $w_j \in (0, 1)$  to labeler  $U_j$  and crowd labels  $l_{i,j}$ . In the strategy of crowd soft labels, the certainty of crowd labels of task  $x_i$  that label type is binary is defined by Laplace estimates in Eqs. (1) and (2). If the labelers with high confidence values to the label, the soft label will have high value. For the case of binary classification, crowd soft labels contain positive soft labels and negative soft labels denoted by  $l_i^+ \in (0, 1)$  and  $l_i^- \in (0, 1)$  respectively.

$$l_i^+ = \frac{\sum_{j=1}^J w_j \cdot \delta(l_{i,j}, +) + 1}{\sum_{j=1}^J w_j \cdot \delta(l_{i,j}, +) + \sum_{j=1}^J w_j \cdot \delta(l_{i,j}, -) + 2}, \quad (1)$$

$$l_i^- = \frac{\sum_{j=1}^J w_j \cdot \delta(l_{i,j}, -) + 1}{\sum_{j=1}^J w_j \cdot \delta(l_{i,j}, +) + \sum_{j=1}^J w_j \cdot \delta(l_{i,j}, -) + 2}, \quad (2)$$

where  $J$  is the total number of labelers,  $w_j$  is the weight of the  $j$ th labeler,  $l_{i,j}$  is the label of item  $\mathbf{x}_i$  given by labeler  $U_j$ , and  $\delta(\cdot)$  is an indicator function whose output will be 1 if two parameters are identical.

Uncertainty (or certainty in other words) is the important information in parameter estimation. In DEW, we propose four uncertainty weighted strategies  $w-f$ ,  $w-b$ ,  $w-pf$  and  $w-pb$  to reduce the bias and roughness. Let  $l_i^m = \max\{l_i^+, l_i^-\}$  be the majority soft label used in our strategies. We first propose a frequency-based weighted strategies denoted by  $w-f$ . Unlike majority voting,  $w-f$  uses  $l_i^m$  to present the multiple label set directly. Besides, it assigns a weight for each instance. The weight is the value of majority soft label  $l_i^m$  in terms of Eq. (3)

$$W_I = \begin{cases} l_i^+ & l_i^+ \geq l_i^- \\ l_i^- & l_i^+ < l_i^- \end{cases} \quad (3)$$

Such certainty estimation is still a little rough and inaccurate, as the estimation is done given limited labels. To estimate it more accurately, we apply a Bayesian estimation to measure the certainty of the class that the instance belongs to. Given  $l_i^+$  positive soft label,  $l_i^-$  negative soft label, and  $M = \{|L_i| | l_{i,j} \neq -1\}$  the size of labels for item  $X_i$ , the posterior probability  $P(y | l_i^+ \cdot M, l_i^- \cdot M)$  follows a Beta distribution  $B(l_i^+ \cdot M + 1, l_i^- \cdot M + 1)$ . The regularized incomplete beta function is given by Eq. (4).

$$I_v(\alpha, \beta) = \sum_{j=\alpha}^{\alpha+\beta-1} \frac{(\alpha + \beta - 1)!}{j!(\alpha + \beta - 1 - j)!} v^j (1 - v)^{\alpha+\beta-1-j} \quad (4)$$

$$W_I = 1 - \min\{I_{0.5}(\alpha, \beta), 1 - I_{0.5}(\alpha, \beta)\} \quad (5)$$

With the decision threshold  $v(0.5)$ , we set the level of certainty as Eq. (5) and propose a Beta-based weighted strategy called  $w-b$ , which assigns the  $l_i^m$  as the integrated crowded label and assigns certainty score as the instance weight.

Pairwise solution is another useful method for integrating multiple labels, which considers both the majority label’s uncertainty and that of the minority label. The basic idea is to generate a couple of weighed pairwise instances, while we assign two sets of weights for the pairwise instances respectively. Here we propose two weighted strategies called *w-pf* and *w-pb*. For *w-pf*, the integrated crowd label of the positive instance  $x_{P_i}$  has been assigned as +, and the weight  $W_{P_i}$  is the  $l_i^+$ , and vice versa. For *w-pb*, the weights via Beta distribution can be calculated from:

$$\begin{aligned} W_{P_i} &= I_{0.5}(\beta, \alpha) \\ W_{N_i} &= 1 - I_{0.5}(\beta, \alpha) \end{aligned} \quad (6)$$

where  $\alpha = l_i^+ \cdot M + 1, \beta = l_i^- \cdot M + 1$ .

Differential Evolution (DE) is a powerful population-based model for optimizing real parameters or real valued functions. In many field of machine learning, DE has been proved efficient to find good solutions for optimization problems. The objective of DE model is to evolve a population of parameter vectors, called individuals, towards the global optimization. The performance of DE depends on three main steps: Mutation, Crossover and Selection, as well as the control parameters like population size  $N_p$ , mutation scaling factor  $F$  and cross rate  $C_r$ .

We propose to use DE to learn optimal weights corresponding to each labeler’s quality for crowdsourcing learning. In our solution, individuals present weight vector  $\mathbf{w}$  with different set of values (i.e., candidates). Each individual in DEW carries fixed training instances  $X$  and the multiple label set  $L$ . Algorithm 1 reports the details of the proposed DEW, which is described as follows:

---

**Algorithm 1.** DEW (weighted consensus algorithm via Differential evolution)

---

**Input:**

Maximum Generation  $T_{max}$ ; Label Integration Strategy  $s$ ;  
Training Crowdsourcing set  $C^a = \{X, Y, L\}$ ; Test Crowdsourcing set  $C^b$ ;

**Output:**

the built classifier  $\Gamma$ ;

- 1:  $\mathcal{W} \leftarrow$  Initialize the  $w_{ij}$  value of  $\mathbf{w}_i$  for each individual using a uniformly random distributed between  $(0, 1)$ .
  - 2: **while**  $t \leq T_{max}$  **do**
  - 3:    $C^a \leftarrow$  Apply the sequence of  $\mathbf{w}_i^t$  to the whole training crowdsourcing set  $C^a$  and update the integrated labels with the selected label integration strategy  $s$ .
  - 4:    $f[\mathbf{w}_i^t] \leftarrow$  Classify the renewed crowdsourcing set  $C^a$  and calculate the fitness of  $\mathbf{w}_i^t$ .
  - 5:    $\mathbf{w}_c^t \leftarrow$  Find the  $\mathbf{w}_c^t$  with the best fitness in  $\mathcal{W}$  according to the value of each  $f[\mathbf{w}_i^t]$ .
  - 6:   **for all** each  $\mathbf{w}_i^t$  in  $\mathcal{W}^t$  **do**
  - 7:      $\mathbf{u}_i^{t+1} \leftarrow$  Perform mutation and crossover according to the Eqs. (8) and (9) on  $\mathbf{w}_i^t$ .
  - 8:      $\mathbf{w}_{i+1}^t \leftarrow$  Apply  $\mathbf{u}_{i+1}^t$  to  $\mathbf{w}_i^t$  if offspring have lower entropy in  $t + 1$  generation.
  - 9:   **end for**
  - 10: **end while**
  - 11:  $\Gamma \leftarrow$  Builds a base classifier  $\Gamma$  on  $\mathbf{w}_c$  and crowdsourcing set  $C^a$  to predict the underlying class label in test crowdsourcing set  $C^b$ .
- 

**Initialization.** During the initialization phrase, we generate a set of  $N_p$  weight candidates:  $W = \{\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_{N_p}\}$ , where  $\mathbf{w}_i = \{w_{i,1}, \dots, w_{i,j}, \dots, w_{i,J}\}$

(i.e., a weight value vector with  $w_{i,j}$  representing the weight value for the  $j$ th labeler). To generate random weights for all individuals, we select each  $w_{i,j}$  value as a uniformly distributed random variable within range  $(0,1)$ .

**Calculation of Fitness Function.** As the true label of item  $x_i$  is unknown in crowdsourcing system, it is necessary to design specific fitness function in our DEW to assess performance of different weight vector  $\mathbf{w}_i$  corresponding to  $J$  labelers. we design the fitness function of entropy-based selection strategy. The fitness of the  $i$ th individual of the  $t$ th generation  $\mathbf{w}_i^t$  can be obtained by applying the weight vector  $\mathbf{w}_i^t$  and crowdsourcing data sets to a learning model. It can be achieved through the following two major steps: (1) using the label integration methods to infer crowd labels for the purpose of obtaining single integration label. (2) training a learning model from the training instances and corresponding integration labels and then evaluating the performance on the whole test data sets using the defined fitness function in Eq. (7).

The performance of entropy on the whole test instances could reflect certainty of the class that the instance belongs to. We minimize the following fitness function using the binary entropy function.

$$F_E = - \sum_{i=1}^n (p_i \cdot \log(p_i) + (1 - p_i) \cdot \log(1 - p_i)), \quad (7)$$

where  $p_i$  and  $(1 - p_i)$  is the probability of instance  $x_i$  being classified as positive or negative by model respectively.

**Mutation and Crossover.** After initialization, DE uses the differential mutation operation to produce the new mutant vector  $\mathbf{v}_i^{t+1}$  with respect to the individuals in the  $t$ th generation  $\mathbf{w}^t$ . For any individual  $\mathbf{w}_i^t$  from the  $t$ th generation, the new mutant vector can be generated as follow:

$$\mathbf{v}_i^{t+1} = \mathbf{w}_{r1^i}^t + F \cdot (\mathbf{w}_{r2^i}^t - \mathbf{w}_{r3^i}^t), \quad (8)$$

where  $F$  is the mutation scaling factor, indicators  $r1^i, r2^i, r3^i$  are mutually exclusive integers randomly chosen from the range  $[1, N_p]$ , which are different from the index  $i$ .

After the mutation stage, a crossover operation is applied to each pair of the target vector  $\mathbf{w}_i^t$  and its corresponding mutation vector  $\mathbf{v}_i^{t+1}$ , which forms the final trail vector  $\mathbf{u}_i^{t+1}$ .

$$u_{ij}^{t+1} = \begin{cases} v_{ij}^{t+1}, & \text{if } \text{rand}(0,1) \leq C_r \text{ or } j = j_{rand}, \\ w_{ij}^t, & \text{otherwise} \end{cases} \quad (9)$$

where  $\text{rand}(0,1)$  represents a uniformly distributed random variable within the range  $[0,1)$ , which is generated for each  $j$ th component of the  $i$ th parameter vector.  $j_{rand} \in [1, 2 \dots, J]$  is a randomly chosen index, which ensures that  $\mathbf{u}_i^{t+1}$  maintains the experiment of at least one labeler from  $\mathbf{v}_i^{t+1}$ .

**Selection.** To determine whether a trail individual  $\mathbf{u}_i^{t+1}$  can replace a target individual vector  $\mathbf{w}_i^t$ , as a new individual  $\mathbf{w}_i^{t+1}$  for the  $t + 1$ th generation, DEW adopts a greedy search strategy. The target individual  $\mathbf{w}_i^t$  is replaced by  $\mathbf{u}_i^{t+1}$ , if  $\mathbf{u}_i^{t+1}$ 's fitness score is better than that of  $\mathbf{w}_i^t$ .

### 3 Experiments and Results

The purpose of this section is to validate the effectiveness of our proposed DEW with four weighted strategies DEW-f, DEW-b, DEW-pf and DEW-pb. For comparison purpose, we compare our algorithms with the existing baseline algorithms: MV, ZC, RY, DS, MV-Freq, MV-Beta, Paired-Freq and Paired-Beta. Due to the limited space, in our experimental tables we denote MV-Freq, MV-Beta, Paired-Freq and Paired-Beta as MVF, MVB, PF and PB respectively.

- MV: majority vote.
- ZC: a probability model to estimate the reliability of labelers [3].
- RY: a Bayesian estimation method containing two parameters sensitivity and specificity [7, 8].
- DS: an integrated method based on maximum likelihood estimation [2].
- MV-Freq: a frequency based majority voting strategy [9].
- MV-Beta: a majority voting strategy based on Beta distribution [9].
- Paired-Freq: a frequency based pairwise strategy [9].
- Paired-Beta: a pairwise strategy based on Beta distribution [9].

In DEW, all setting for parameters are the same ( $N_p$ ,  $F$ ,  $C_r$ , and  $T_{max}$  are set to 50, 0.5, 0.9 and 100 respectively). Naive Bayes is the base classifier  $\Gamma$  in the evolutionary phase. All results are obtained via 5 runs of 10-fold cross-validation.

We run our experiments on 10 binary classification benchmark data sets from UCI data repository [11]. As benchmark data sets are designed for traditional classification, in our experiments, in order to simulate a crowdsourcing process to obtain multiple labels of each instance, we first hide the original true labels of all instances. Then, we employ 4 simulated labelers who possess different level of reliability to label each instance. More specifically, for each labeler, the crowd label which belongs to the true label is assigned to each instance with probability  $p_j$  and the opposite value was assigned with probability  $1 - p_j$ . In the series of experiments, the labeling quality of each labeler was generated randomly from a uniform distribution on the interval [0.5, 0.8].

The detailed experimental results are presented in Tables 1, 2, 3 and 4. Besides, the averaged accuracies of all algorithms on 10 datasets are summarized at the bottom of the table. The averaged accuracies of DEW-f, DEW-b, DEW-pf and DEW-pb are 79.82%, 79.39%, 77.75%, and 80.02%, respectively, which are much higher than that of MV (71.84%). These results illustrate that the performance of DEW, including DEW-f, DEW-b, DEW-p, and DEW-pf, are all significantly better than that of the MV method.

**Table 1.** Results of group (a) on classification accuracy (%).

Dataset	MV	ZC	RY	DS	MVF	DEW-f
biodeg	66.04	71.22	71.82	66.26	77.14	81.86
breast-c	67.84	66.6	57	70.17	70.17	70.24
breast-w	86.89	89.68	89.27	65.56	91.56	92.74
credit-a	76.58	77.16	77.28	55.51	82.81	83.54
credit-g	61.7	65.36	68.04	70	69.22	71.54
diabetes	65.79	68.48	71.2	65.07	65.78	71.89
heart-s	65.56	67.33	65.85	55.56	74.89	75.85
hepatitis	68.93	71.3	79.67	79.2	81.37	82.63
horse-colic	80.44	79.26	76.87	63.09	81.46	83
ionosphere	78.62	78.05	73.34	64.09	81.41	84.89
<b>Average</b>	71.84	73.44	73.03	65.45	77.58	79.82

**Table 2.** Results of group (b) on classification accuracy (%).

Dataset	MV	ZC	RY	DS	MVB	DEW-b
biodeg	66.04	71.22	71.82	66.26	72.94	79.03
breast-c	67.84	66.6	57	70.17	70.1	70.24
breast-w	86.89	89.68	89.27	65.56	91.69	93.32
credit-a	76.58	77.16	77.28	55.51	81.22	84.03
credit-g	61.7	65.36	68.04	70	69.1	70.06
diabetes	65.79	68.48	71.2	65.07	68.28	72.74
heart-s	65.56	67.33	65.85	55.56	74.07	77.26
hepatitis	68.93	71.3	79.67	79.2	78.27	82.3
horse-colic	80.44	79.26	76.87	63.09	81.7	82.62
ionosphere	78.62	78.05	73.34	64.09	81.65	82.33
<b>Average</b>	71.84	73.44	73.03	65.45	76.90	79.39

**Table 3.** Results of group (c) on classification accuracy (%).

Dataset	MV	ZC	RY	DS	PF	DEW-pf
biodeg	66.04	71.22	71.82	66.26	73.51	78.82
breast-c	67.84	66.6	57	70.17	70.17	70.17
breast-w	86.89	89.68	89.27	65.56	91.23	92.29
credit-a	76.58	77.16	77.28	55.51	84.41	84.46
credit-g	61.7	65.36	68.04	70	70	69.56
diabetes	65.79	68.48	71.2	65.07	65.07	65.07
heart-s	65.56	67.33	65.85	55.56	76.3	76.3
hepatitis	68.93	71.3	79.67	79.2	83.33	83.1
horse-colic	80.44	79.26	76.87	63.09	81.34	82.85
ionosphere	78.62	78.05	73.34	64.09	74.92	74.92
<b>Average</b>	71.84	73.44	73.03	65.45	77.03	77.75

**Table 4.** Results of group (d) on classification accuracy (%).

Dataset	MV	ZC	RY	DS	PB	DEW-pb
biodeg	66.04	71.22	71.82	66.26	81.03	81.49
breast-c	67.84	66.6	57	70.17	70.17	70.17
breast-w	86.89	89.68	89.27	65.56	91.14	92.51
credit-a	76.58	77.16	77.28	55.51	83.71	83.25
credit-g	61.7	65.36	68.04	70	69.96	71.16
diabetes	65.79	68.48	71.2	65.07	65.07	70.04
heart-s	65.56	67.33	65.85	55.56	76.22	76.15
hepatitis	68.93	71.3	79.67	79.2	83.1	84.4
horse-colic	80.44	79.26	76.87	63.09	81.46	83.71
ionosphere	78.62	78.05	73.34	64.09	81.47	87.34
<b>Average</b>	71.84	73.44	73.03	65.45	78.33	80.02

**Table 5.** The Wilcoxon test for group (a).

Algorithm	MV	ZC	RY	DS	MVF	DEW-f
MV	–					
ZC	•	–				
RY			–			
DS				–		
MV-Freq	•	•	•	•	–	
DEW-f	•	•	•	•	•	–

**Table 6.** The Wilcoxon test for group (b).

Algorithm	MV	ZC	RY	DS	MVB	DEW-b
MV	–					
ZC	•	–				
RY			–			
DS				–		
MV-Beta	•	•	•	•	–	
DEW-b	•	•	•	•	•	–

**Table 7.** The Wilcoxon test for group (c).

Algorithm	MV	ZC	RY	DS	PF	DEW-pf
MV	–					
ZC	•	–				
RY			–			
DS				–		
Paired-Freq	•		•	•	–	
DEW-pf	•	•	•	•		–

**Table 8.** The Wilcoxon test for group (d).

Algorithm	MV	ZC	RY	DS	PB	DEW-pb
MV	–					
ZC	•	–				
RY			–			
DS				–		
Paired-Beta	•	•	•	•	–	
DEW-pb	•	•	•	•	•	–

Then, we complete the Wilcoxon signed-ranks test for comparing each pair of algorithms. Tables 5, 6, 7 and 8 show the detailed comparison results of the Wilcoxon test. In these Tables, • indicates that the algorithm in the row significantly outperforms the algorithm in the corresponding column. According to these summaries of the Wilcoxon test, DEW-f, DEW-b, DEW-pb significantly outperform MV-Freq, MV-Beta, and Paired-Beta, respectively. DEW-pf is inferior to Paired-Freq. Besides, DEW significantly outperforms MV, ZC, RY, and DS.

From these experimental results, we can see that DEW significantly outperforms other consensus methods. The weighted soft majority voting strategy that utilizes the quality of labelers to the process of integrating labels is effective. The differential evolution-based weighted approach is an efficient method for solving the problem of estimating labeler quality.

## 4 Conclusions and Future Work

We propose to improve crowdsourcing learning by optimizing the weight vector of labelers via differential evolution. As the reliability of labelers is various, many existing works use Expectation-Maximization (EM) to improve that. In this paper, we investigate the existing labeler weighted methods and propose that evolution-based optimizing approach is a good solution to the problem of estimating labeler quality. Our approach, DEW, uses differential evolution (DE) to search optimal weights for multiple labelers. The experimental results show that DEW can significantly improve data and model quality.

For task type and task reward amount based trust issues, researchers addressed them by population-based optimization algorithms, such as NSGA-II [12]. We believe that considering more factors to make quality estimation can improve their performance. This is the main direction for our future work.



## References

1. Bazi, Y., Alajlan, N., Melgani, F., AlHichri, H., Malek, S.: Differential evolution extreme learning machine for the classification of hyperspectral images. *IEEE Geosci. Remote Sens. Lett.* **11**(6), 1066–1070 (2014)
2. Dawid, A.P., Skene, A.M.: Maximum likelihood estimation of observer error-rates using the EM algorithm. *Appl. Stat.*, 20–28 (1979)
3. Demartini, G., Difallah, D.E., Cudr-Mauroux, P.: ZenCrowd: leveraging probabilistic reasoning and crowdsourcing techniques for large-scale entity linking. In: *Proceedings of the 21st International Conference on World Wide Web*, pp. 469–478. ACM (2012)
4. Georgescu, M., Zhu, X.: Aggregation of crowdsourced labels based on worker history. In: *Proceedings of the 4th International Conference on Web Intelligence, Mining and Semantics*, pp. 1–11. ACM (2014)
5. Howe, J.: The rise of crowdsourcing. *Wired Mag.* **14**(6), 1–4 (2006)
6. Maulik, U., Bandyopadhyay, S., Saha, I.: Integrating clustering and supervised learning for categorical data analysis. *IEEE Trans. Syst. Man Cybern. Part A Syst. Hum.* **40**(4), 664–675 (2010)
7. Raykar, V.C., Yu, S., Zhao, L.H.: Supervised learning from multiple experts: whom to trust when everyone lies a bit. In: *Proceedings of the 26th Annual International Conference on Machine Learning*, pp. 889–896. ACM (2009)
8. Raykar, V.C., Yu, S., Zhao, L.H.: Learning from crowds. *J. Mach. Learn. Res.* **11**, 1297–1322 (2010)
9. Sheng, V.S.: Simple multiple noisy label utilization strategies. In: *2011 IEEE 11th International Conference on Data Mining*, pp. 635–644. IEEE (2011)
10. Whitehill, J., Wu, T.F., Bergsma, J., Movellan, J.R., Ruvolo, P.L.: Whose vote should count more: optimal integration of labels from labelers of unknown expertise. In: *Advances in Neural Information Processing Systems*, pp. 2035–2043 (2009)
11. Witten, I.H., Frank, E., Hall, M.A.: *Data Mining: Practical Machine Learning Tools and Techniques*, 3rd edn. Morgan Kaufmann, San Francisco (2011)
12. Ye, B., Wang, Y., Liu, L.: Crowd trust: a context-aware trust model for worker selection in crowdsourcing environments. In: *ICWS*, pp. 121–128. IEEE (2015)
13. Zhang, Y., Chen, X., Zhou, D., Jordan, M.I.: Spectral methods meet EM: a provably optimal algorithm for crowdsourcing. In: *NIPS*, pp. 1260–1268 (2014)



# A Search Optimization Method for Rule Learning in Board Games

Hui Wang<sup>1</sup>, Yanni Tang<sup>1</sup>, Jiamou Liu<sup>2</sup>, and Wu Chen<sup>1,3</sup>(✉)

<sup>1</sup> College of Computer and Information Science, Southwest University,  
Chongqing 400715, China

<sup>2</sup> The University of Auckland, Auckland, New Zealand  
jiamou.liu@auckland.ac.nz

<sup>3</sup> Institute of Logic and Intelligence, Southwest University,  
Chongqing 400715, China  
chenwu@swu.edu.cn

**Abstract.** A general game playing (GGP) system aims to play previously unknown board games with changeable rules without human intervention. Taking changeable game rules into consideration, a game description language presents formal descriptions of a game. Based on this description, legal moves can be automatically generated so that each player in GGP system only needs to solve the problems of *searching* and *learning* for playing well. Traditional search methods demand the player to compute all legal moves, which can be very time consuming. In GGP, the coordinate of cells in the game board is very important in board game rules. Thus we address the relationship among cell coordinates. Borrowing an idea from rule learning to prune the board game search tree, we propose a new search optimization method to reduce running time when searching a large search space. We further prove that this method can effectively improve the searching efficiency through a comparative experiment with Gomoku game in GGP system.

**Keywords:** General game playing · Game description language  
Rule learning · Searching optimization

## 1 Introduction

One of the most prominent successes of artificial intelligence is Deep Blue's defeat of world chess champion Garry Kasparov. However, a fundamental limitation exists with Deep Blue as the game rules are built into the system that prevents the system from achieving a desirable level of machine intelligence. As it's not practical to construct specific strategic engines for all games, GGP emerged as a promising direction which aims for an intelligent system that automatically learns games rules and derives game strategies without human intervention [1]. A GGP sets itself apart from the traditional game program in that it implements

an interpreter program [2], which takes as input formal descriptions of a game – as specified in a game description language (GDL) [3] – and generate legal moves under a specific state with an interpreter program.

Since a GGP platform allows us to play all kinds of board games, how to play these games well in GGP becomes an important issue. Numerous studies have been done to assist players on a specific game, such as AlphaGo [4]. But in general games, there are still many problems to solve. Since in GGP, games are described as rules, and the result of rule learning is also a set of rules. Rules learning from game records can be regarded as decision strategies and searching strategies. Therefore, this paper combines the knowledge of GGP with rule learning to try to find out a searching optimization method. To make it simple, we choose Gomoku game as our example. We introduce an experiment of playing Gomoku game with two players, in which players have the same decision strategies but one applied the rule obtained from training, and the other did not use it. The results show that the player who uses the search rules can more quickly find the best move in large scale games with an even win rate.

Our contributions can be summarized as follows:

1. This paper presents a searching optimization method that applies rule learning to GGP field. The idea is evidenced by game Gomoku. An algorithm of generating searching rules is illustrated based on *Inductive Logic Programming*.
2. Knowledge obtained from rule learning is formalized as game playing rules. It's important to study the relation between game playing rules and game rules, the method of this paper provides a basement to study this relationship further.

The paper is organized as follows. Section 2 illustrates related work. Section 3 recalls the basic concepts of GGP and rule learning. Section 4 introduces the rule learning process of searching rules of Gomoku game based on *Inductive Logic Programming* in GGP. Section 5 presents a comparative experiment with Gomoku in GGP. Section 6 concludes the paper and discusses on future work.

## 2 Related Work

Learning about general concepts or rules has been a crucial research problem in artificial intelligence. *Inductive Logic Programming* is an important paradigm to cope with the issue of rule learning. Arindam Mitra et al. addressed a question-answering challenge by combining statistical methods with inductive rule learning and reasoning [5]. Furthermore, David Garcíá et al. proposed an interpretability improvement for fuzzy rule bases obtained by the iterative rule learning approach [6].

More recently, Ondřej Kuželka et al. introduced a setting for learning possibilistic logic theories from defaults of the form “if alpha then typically beta” [7]. They aim at studying the problem of reasoning with default rules from a machine learning perspective.

Specifically on GGP, there exist also numerous works that made great progress. Günther et al. proposed a search algorithm that exploits this information in single-player games [8]. Moreover, Dave and Zhang proposed a lifted backward search in GGP system [9].

Different from the methods we mentioned above, we refer to the way proposed by Krajnanský et al. [10] based on our requirements to generate simple rules to minimize search space.

### 3 Preliminaries

#### 3.1 General Game Playing and Game Description Language

General game players are systems who are able to accept descriptions of arbitrary games at runtime and derive corresponding strategies to play those games effectively without human intervention. In other words, they do not know the rules until the games start [11, 12]. In GGP structure, the game manager is at the center of the ecosystem. There are databases for game descriptions, match histories, and temporary states histories during game playing. The game manager interacts with game players through TCP/IP protocol.

Every finite game can be modeled as a state transition system. A game description is a finite collection of rules and a GDL is a specific language to describe this collection [13]. A basic game description includes the specific attributes and relations. The reader is referred to [3] for numerous examples and tutorials.

#### 3.2 Rule Learning

*Rule learning* is a process of generating a set of rules to evaluate unknown instances from training data set [14]. A formalized rule form is  $\oplus \leftarrow f_1 \wedge f_2 \wedge \dots \wedge f_i \dots \wedge f_L$ , where the right hand side  $f_1 \wedge f_2 \wedge \dots \wedge f_i \dots \wedge f_L$  is called *rule body* and represents this rule's preconditions, while the left part  $\oplus$  is called *rule head* which represents the result of this rule. The rule body is a conjunction that consists of a sequence of literals  $f_i$ . The symbol  $\wedge$  means conjunction. Every literal  $f_i$  is a boolean expression to examine the attributions/properties of examples. Suppose we have two rules:  $\oplus \leftarrow f_1 \wedge f_2$ ; and  $\oplus \leftarrow f_3 \wedge f_4$ . The first rule means that an example is positive if it is covered by properties  $f_1 \wedge f_2$ . The second rule means that an example is positive if it is covered by properties  $f_3 \wedge f_4$ . This could also be combined to write as a single disjunctive rule  $\oplus \leftarrow (f_1 \wedge f_2) \vee (f_3 \wedge f_4)$ .

### 4 Learning Search Rules

Since a GGP system can record game history, one can easily obtain a training set from the game records. Many studies exist, e.g. [15], that provide efficient instance selection algorithms to reconstruct training sets. In order to distinguish the samples in the training set, these samples should be divided into two parts,

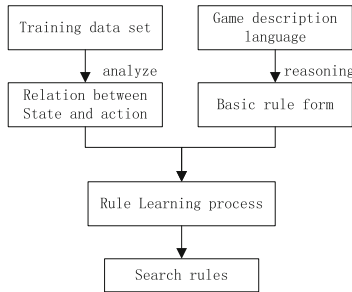
we let  $E^+$  and  $E^-$  to represent these two parts respectively. In order to define a framework for our GGP solution, a general game state transition model for games can be simply regarded as sets of state-action pairs  $(s, a)$  [16]. Formally,

$$E^+ = \{(s, a) | s \in S^*, a \in A^+\} \quad \text{and} \quad E^- = \{(s, a) | s \in S^*, a \in A^-\} \quad (1)$$

where  $S^*$  represents the set of history states,  $A^+$  ( $A^-$ ) represents the set of actions which are (not) most possible to be applied in  $s$ ,  $A^+$  ( $A^-$ ) can make positive (negative) effect on winning,  $A^+ \subseteq A$  and  $A^- \subseteq A$ .  $A^+ \cap A^- = \emptyset$  and  $A^+ \cup A^- = A$ . The state  $s'$  is the outcome after applying  $a$  to  $s$ , where  $s \subseteq S^*$  and  $s' \subseteq S^*$ .  $E^+$  is called *good* examples set,  $E^-$  is called *bad* examples set.

Our purpose is to get a formal representation  $R$  of rules by rule learning. The representation  $R$  should allow us to reduce the search space. The rule is generated from covering all samples of the same domain  $D$ . In order to ensure that  $R$  can cover all good examples and try its best to exclude bad examples, in any state  $s$ , we should include all possible action  $a$  and avoid any impossible actions. In order to find  $R$ , here are some problems we need to solve: **(a)** What kind of representation can describe  $R$ ; **(b)** What “covering” a state-action pair  $(s, a)$  means; **(c)** How  $R$  can be learned.

In fact, Michal et al. presented their answer in [10]. However, in GGP, things are different. In our work, we refer to GDL since GDL provides us the game rules collection. As for board games, such as Tic-Tac-Toe, Connect Four, Hex, the coordinates of cells are the most important part in their GDL description: States of board games are described in different cells’ states (coordinates, marked or not, by whom. e.g. (cell 1 1 x)); actions are also described using coordinates (e.g. mark(1 1)). So the coordinates of different cells are the key to represent the basic rule form. Thus, we propose a model to process this in Fig. 1:



**Fig. 1.** Model of generating search rules with game description

In a Gomoku game, our example, we can easily find that the game goal description is to construct a line with 5 pieces. The player firstly needs to construct a line with 2 pieces, then three then four and lastly five pieces to reach the termination of the game. In the other hand, the player must stop the opponent from constructing a line with pieces. Thus, either we’d like to attack or defense, we both firstly need to search within the places next to our or opponents pieces.

Through the analysis and reasoning above, we propose an algorithm for learning searching rules in board games as follows:

---

**Algorithm 1.** Algorithm of generating searching rules based on board coordinate

---

**Input:**

The set of *good* (*bad*) examples,  $E^+$  ( $E^-$ ); Search space  $\omega$ ; Whole board space  $\Omega$ ;

**Output:**

The set of searching rules  $R$ ;

- 1:  $r_1 : \omega = \Omega \leftarrow s = \emptyset$ ;  $\triangleright s = \emptyset$  means game starting state.
  - 2: **for** each (s,a) in  $E^+$  **do**
  - 3:  $s = \{(m_1, n_1), (m_2, n_2), \dots, (m_k, n_k), \dots, (m_j, n_j), 1 \leq k \leq j\}$ ;
  - 4:  $a = \text{move}(m, n)$ ;
  - 5:  $r_2 : \omega = \Omega \cap \bigcup_{k=1}^j \{\text{move}(m_k \pm \sigma, n_k) \cup \text{move}(m_k, n_k \pm \sigma) \cup \text{move}(m_k \pm \sigma, n_k \pm \sigma)\} \leftarrow \sigma = \min\{N | \exists \text{move}(m_k, n_k) \text{ s.t. } |m_k - m| \leq N \wedge (|n_k - n| \leq N)\}$ ;
  - 6: **end for**
  - 7: **for** each (s,a) in  $E^-$  **do**
  - 8:  $s = \{(m_1, n_1), (m_2, n_2), \dots, (m_k, n_k), \dots, (m_j, n_j), 1 \leq k \leq j\}$ ;
  - 9:  $a = \text{move}(m, n)$ ;
  - 10:  $M = \{a\}$ ;
  - 11:  $r_3 : \{\omega = \omega \leftarrow M \subsetneq s\}$ ;
  - 12:  $r_4 : \{(\omega = \omega - M \leftarrow M \subseteq s) \wedge (\omega = \omega + M \leftarrow \text{after searching under this state } s)\}$ ;
  - 13: **end for**
  - 14:  $R = \{r_1, r_2, r_3, r_4\}$ ;
  - 15: **return**  $R$ ;
- 

In Algorithm 1, we can see, when game board is empty, we use  $r_1$ , i.e., searching the whole board. Then in *good* examples using *Inductive Logic Programming* to find the relationship between the coordinate of action and the existing coordinates of game state. This relationship is described as  $r_2$ . Based on  $r_2$ , check if  $r_2$  also covers *bad* examples, if not, i.e.,  $M \subsetneq s$ , then search space keep the same ( $\omega = \omega$ ), written as  $r_3$ . Otherwise, exclude this *bad* example ( $\omega = \omega - M$ ). The computational complexity is  $O(n^2)$ , where  $n$  is the number of training examples.

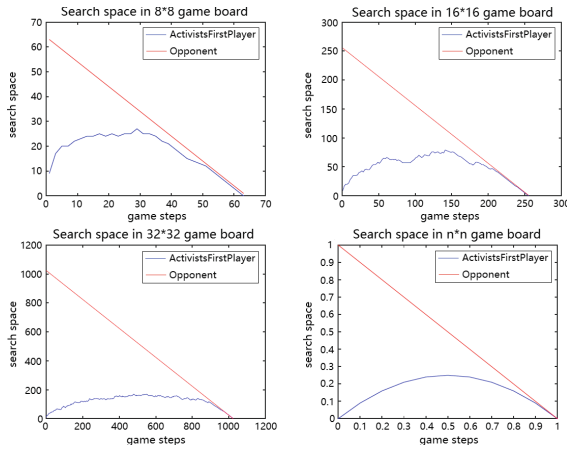
## 5 Experiment Results

According to Algorithm 1, we learn searching rules based on the training set in Gomoku. We get the result that  $\sigma = 1$ , the search space is  $\omega = \Omega \cap \bigcup_{k=1}^j \{\text{move}(m_k \pm 1, n_k) \cup \text{move}(m_k, n_k \pm 1) \cup \text{move}(m_k \pm 1, n_k \pm 1)\}$ , which means we should search the places just next to the pieces existing in the board. We call these places (legal moves) as *activists* positions. The result holds the same information as what we analyzed and reasoned above.

In our implementation, for instance, we know X player has occupied position (4, 4), that Player O has occupied position (3, 3), and now it's X player's turn.

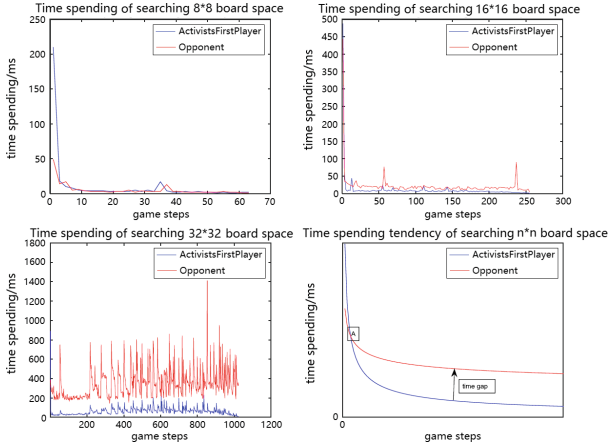
According to the rules learned before, the next move is most probably to mark one of the positions in the set of  $\{(2,2), (2,3), (2,4), (3,2), (3,4), (3,5), (4,2), (4,3), (4,5), (5,3), (5,4), (5,5)\}$ . Note that in the experiment we should return a legal move instead if there is no expected move in the activists set.

We create a player based on this searching method with the same playing strategies as opponent's (always get a draw), which is called ActivistsFirstPlayer. Obviously, the player's name means searching the activists list first to get expected results. The opponent use traditional ways like searching in all legal moves list. There are three games with a size of  $8 \times 8$ ,  $16 \times 16$  and  $32 \times 32$  board respectively. Obviously, from Fig. 2, two players have different search spaces, the search space of ActivistsFirstPlayer increases from 0, and then goes down, while the opponent's decreases linearly by the board size. More importantly, in every step, the search space of ActivistsFirstPlayer stays below the opponent's.



**Fig. 2.** Search space of different size game boards

Since ActivistsFirstPlayer only needs to search through a smaller space to find the next move, the time spending of ActivistsFirstPlayer on calculate is evidently less than the opponent's. However, ActivistsFirstPlayer will cost some time to get such a set of moves. From the experiment results of time spending in Fig. 3, it is worthy to do this job, especially to search a great scale area. In addition, we can decrease this cost to a smaller level by several ways comparing with traversing the whole area. On the  $8 \times 8$  board, the cost of time of both players has little difference. In fact, ActivistsFirstPlayer pays little time to search but spends extra time on constructing the activists list, because the list is always dynamically changing. In the  $16 \times 16$  board, opponent spends several times time than ActivistsFirstPlayer for each move. And in the  $32 \times 32$  board, the time gap between two payers' time spending is getting bigger and bigger. The results prove that this method is applicable for the games and which can also reduce search space through rule learning, rather than a simple statistic machine learning.



**Fig. 3.** Time spending on searching different size game boards

Based on these three real experiments outcomes, we can simulate the time spending while searching  $n \times n$  board space. In Fig. 3, at the beginning, tradition search method spends less time than the ActivistsFirstPlayer. The reason is that ActivistsFirstPlayer needs to spend extra time on constructing the activists list. But this extra time spending will be compensated. In the dot *A* marked in the diagraph, the time spending of both players are the same. Later on, the time spending of ActivistsFirstPlayer is getting less than the tradition method. It is obviously that when the board size becomes bigger and bigger, the dot *A* will get closer to *y*-axis. The time gap will become bigger. Therefore, the experiments results have demonstrated this search optimization method is more efficient while searching a large scale field at a high accurate rate level.

## 6 Conclusion and Future Work

This paper first introduce rule learning into GGP to learn trustworthy rules to determine what should or shouldn't be searched for board games playing. We found the *goal* function written in GDL for the game is more possible to tell players which state can be closer to game termination. We analyzed properties from *goal* function to define the rule form as the input of rule learning process first, and then applied *Inductive Logic Programming* method to learn out searching rules. Last, we applied these rules to the game searching process, and introduced an experiment to prove the accuracy and effectiveness of this new method.

In the future, our purpose is to establish an independent module to generate search rules, and apply these rules to most of board games automatically and effectively. So that this module can be integrated to enrich GGP system with high efficiency. Our method in this paper is practical, but there is no doubt that further work should be done to improve it.



**Acknowledgement.** This work was supported by the Key Project of Chongqing Humanities and Social Science Key Research Base: Research on Coalition Welfare Distribution Mechanism and Social Cohesion Based on Cooperative Game Theory and the Ratification Number is 18SKB047.

## References

1. Genesereth, M., Love, N., Pell, B.: General game playing: overview of the AAI competition. *AI Mag.* **26**(2), 62–72 (2005)
2. Świechowski, M., Mańdziuk, J.: Fast interpreter for logical reasoning in general game playing. *J. Log. Comput.* **26**(5), 1697–1727 (2014)
3. Love, N., Hinrichs, T., Haley, D., et al.: General game playing: game description language specification (2008)
4. Silver, D., Huang, A., Maddison, C.J., et al.: Mastering the game of Go with deep neural networks and tree search. *Nature* **529**(7587), 484–489 (2016)
5. Mitra, A., Baral, C.: Addressing a question answering challenge by combining statistical methods with inductive rule learning and reasoning. In: Proceedings of the Thirtieth AAI Conference on Artificial Intelligence (2016)
6. Garcíá, D., Gañez, J.C., González, A., Peñeza, R.: An interpretability improvement for fuzzy rule bases obtained by the iterative rule learning approach. *Int. J. Approximate Reasoning* **67**, 37–58 (2015)
7. Günther, M., Schiffel, S., Thielscher, M.: Factoring general games. In: Proceedings of the International Joint Conference on Artificial Intelligence-09 workshop on general game playing, pp. 27–34 (2009)
8. Kuželka, O., Davis, J., Schockaert, S.: Learning possibilistic logic theories from default rules. In: Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (2016)
9. de Jonge, D., Zhang, D.: Lifted backward search for general game playing. In: Kang, B.H., Bai, Q. (eds.) *AI 2016. LNCS (LNAI)*, vol. 9992, pp. 3–16. Springer, Cham (2016). [https://doi.org/10.1007/978-3-319-50127-7\\_1](https://doi.org/10.1007/978-3-319-50127-7_1)
10. Krajnanský, M., Hoffmann, J., et al.: Learning pruning rules for heuristic search planning. In: European Conference on Artificial Intelligence, pp. 483–488 (2014)
11. Genesereth, M., Thielscher, M.: *General Game Playing*. Morgan & Claypool Publishers, Williston (2014)
12. Kaiser, D.M.: The design and implementation of a successful general game playing agent. In: Wilson, D., Sutcliffe, G. (eds.) *International Florida Artificial Intelligence Research Society Conference 2007*, pp. 110–115. AAAI Press, California (2007)
13. Thielscher, M.: A general game description language for incomplete information games. In: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 994–999. AAAI Press, Atlanta (2010)
14. Fürnkranz, J., Gamberger, D., Lavrac, N.: *Foundations of Rule Learning. Cognitive Technologies*. Springer, Heidelberg (2012). <https://doi.org/10.1007/978-3-540-75197-7>
15. Liu, C., Wang, W., Wang, M., Lv, F., Konan, M.: An efficient instance selection algorithm to reconstruct training set for support vector machine. In: *Knowledge-Based Systems*, pp. 58–73 (2017)
16. Zhang, D., Thielscher, M.: Representing and reasoning about game strategies. *J. Philos. Log.* **44**(2), 203–236 (2015)



# Image Segmentation Based on MRF Combining with Deep Learning Shape Priors

Yan Wang and Xili Wang<sup>(✉)</sup>

School of Computer Science,  
Shaanxi Normal University, Shaanxi 710119, China  
wangxili@snnu.edu.cn

**Abstract.** In image segmentation tasks, shadow, clutter background and various interference factors in the image increase the difficulty of segmentation, and lead to unsatisfied results. To solve these problems, this paper proposes an image segmentation algorithm combining MRF (Markov Random Field) with deep learning shape priori. The target shape priori information is modelled and generated from deep learning models: Restricted Boltzmann Machine (RBM), Deep Belief Network (DBN), and Deep the Boltzmann Machine (DBM). Then the shape priori information is further defined and added to the energy function of the MRF image segmentation algorithm. Since the shape priori restricts the target shape and restrains the interference factors, better segmentation results are obtained. The proposed method is compared with traditional MRF and some comparable image segmentation methods in two datasets, the experiments results demonstrate the effective of the proposed method.

**Keywords:** Image segmentation · Deep learning · Shape priori  
MRF

## 1 Introduction

Images can not only clearly express the shape and appearance of objects, but also provide the most intuitive experience in human recognition. Image segmentation is one of the most important tasks in computer vision and other fields. However, in practical applications, due to the presence of noise, object occlusion, clutter background, and other interference factors in the images, image segmentation faces many difficulties and challenges. The traditional image segmentation methods [1, 2] are mainly based on the low-level information, such as gray, color, edge, and texture extracted and defined manually. MRF image segmentation method is based on the probability graph model. This method defines the image segmentation problem by a form of energy function. The optimal segmentation result is obtained by solving the minimum of the function. Other constraint information can be easily added to the energy function as new item to achieve more ideal segmentation effect, which makes it possible to introduce shape constraint information into the image segmentation.

Multi-layer structure of deep learning models [3] have the abilities to automatically extract features of complex and large data, represent multi-level features implied in data. They can also be used to describe complex and various target shape, and then

provide shape constraint for subsequent image segmentation. Although there are many handmade methods characterizing two-dimensional shapes, the process of shape modelling is complicated. Compared with these traditional methods, modelling shape by deep learning not only automatically extract the shape features of the training set, but also represent shape flexibly and generate shape conveniently. In the field of image segmentation, using shape prior to constrain the segmentation has been applied to some image segmentation methods [11, 12]. Characterizing shape by deep learning models and converting it into a priori information, then introducing it to MRF image segmentation method is relatively novel and can significantly improve the segmentation results, especially for complicated images.

In this paper, target prior shape is modelled by RBM, DBN and DBM. And then introducing to the MRF image segmentation energy function. The paper is organized as follows: Sect. 2 introduces proposed method. Section 3 is the experimental results and analysis. Section 4 is conclusion.

## 2 Image Segmentation Based on MRF Combining with Deep Learning Shape Priors

### 2.1 MRF Image Segmentation

Given an image  $I$ , image segmentation can be regard as a labeling problem.  $P = \{1, 2, \dots, m\}$  denote all the pixels of the image to be segmented. Considering the background/objective classification problem,  $L = \{0, 1\}$  corresponds to target and background, the label of the target is 1 and the background is marked as 0. For the image  $I$ , solving the image segmentation problem is defined as finding an optimal mapping  $\Phi : P \rightarrow L$  It is equal to maximize posterior probability  $P(f|I)$ , where  $f$  is a set of labels assigned to the image pixels. Thus, the problem of segmentation is transformed into an optimization problem. The optimal  $f$  is solved by

$$f^* = \arg \max_f P(f|I) \quad (1)$$

According to the Hammersley-Chifford [8] theorem, the posterior probability is defined as Eq. 2, where  $Z$  is the normalization factor and  $E(f|I)$  is the energy function.

$$P(f|I) = \frac{1}{Z} e^{-E(f|I)} \quad (2)$$

After adding shape prior, the MRF energy function is defined as two parts as follows, namely  $E^A$  and  $E^S$  in Eq. 3, denoting appearance and shape priori respectively.

$$E(f|I) = E^A(f|I) + E^S(f|I) \quad (3)$$

## 2.2 Appearance Priori

In Eq. 3,  $E^A$  is modeled by the image feature using the combination of first-order and second-order potential functions define as follows.

$$E^A(f|I) = \sum_{p \in P} V_p^A(f_p|I) + \sum_{p \in P} \sum_{q \in N_p} V_{pq}^A(f_p, f_q|I) \quad (4)$$

Where  $P$  is the set of image pixels,  $p$  is the pixel in  $P$ ,  $f_p$  is the label assigned to the pixel  $p$ ,  $N_p$  is the set of neighbor pixels of  $p$ ,  $q$  is the pixel in  $N_p$ ,  $f_q$  is the label assigned to pixel  $q$ . The first-order potential function  $V_p^A(f_p|I)$  represents the possibility that the pixel  $p$  belonged to the class represented by  $f$ . The second-order potential function  $V_{pq}^A(f_p, f_q|I)$  is the cost or penalty, which punishing the irregularity of the adjacent pixels  $p$  and  $q$ .

The appearance (color) models for object and background areas of an image are established respectively. We model the first term of  $E^A$  by Gaussian mixture model. In this paper, the number of component  $k$  is set to 5. For pixel  $p$ , the probability density function in the color space is  $\Pr(d_p | f_p, \mu_k, \sum_k)$ , where  $d_p$  is the color feature of pixel  $p$ . The first-order potential function  $V_p^A(f_p|I)$  in Eq. 4 can be defined in the form of a likelihood function as follows.

$$V_p^A(f_p) = - \sum_k \ln \alpha_k \Pr(d_p | f_p, \mu_k, \sum_k) \quad (5)$$

$\alpha_k$  is the weight of the  $k$ -th Gaussian component in the mixed model,  $\alpha_k \geq 0$ .  $\mu_k$ ,  $\sum_k$  represent the mean and covariance matrix of the  $k$ -th Gaussian component. The parameters of Gaussian distributions are estimated by maximum likelihood method.

The second-order potential function  $V_{pq}^A(f_p, f_q|I)$  is defined as Eqs. 6 and 7.

$$V_{pq}^A(f_p, f_q|I) = \begin{cases} \delta(p, q|I) & \text{if } f_p \neq f_q \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

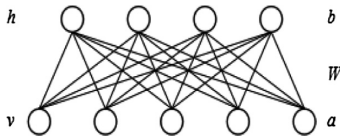
$$\delta(p, q|I) = \lambda \exp \left[ - \frac{(v_p - v_q)^2}{2\sigma^2} \right] \frac{1}{\text{dist}(p, q)} \quad (7)$$

$v_p$  and  $v_q$  represent the feature vector of pixel  $p$  and  $q$ .  $\text{dist}(p, q)$  is the Euclidean distance between  $p$  and  $q$ .  $\sigma$  is the estimation of noise, and the weight  $\lambda$  measures the relative importance of the second order potential function.

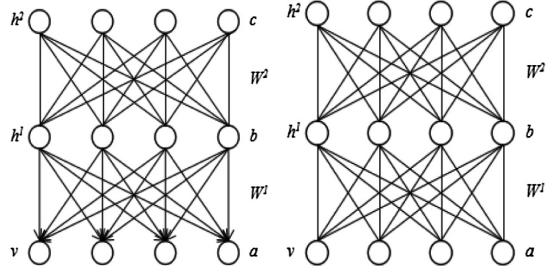
## 2.3 Deep Learning Shape Prior

**Deep Learning Models.** In this paper, target shape is modeled by the deep learning models RBM, DBN [5] and DBM [6] that can express shapes completely and flexibly

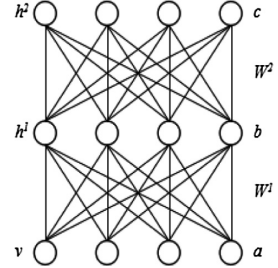
even if the target shapes are complex and changed greatly. The structures of these three Deep learning models are shown as follows (Figs. 1, 2 and 3).



**Fig. 1.** The structure of RBM



**Fig. 2.** The structure of DBN



**Fig. 3.** The structure of DBM

RBM [16], contains a visible layer and a hidden layer. The vector  $v$  of visible layer is the input sample data, and the hidden layer vector represents the shape feature extracted from  $v$ .  $W$  is the weight matrix,  $a$  and  $b$  are the offset vectors between the visible layer and the hidden layer. DBN contains a visible layer and multiple hidden layers. Each hidden layer corresponds to the high-level abstract feature captured from the adjacent low layer, which can better reflect the intrinsic structure and feature of the input data.  $W1$  is the weight matrix of the connection between the visible layer and the hidden layer, while  $W2$  is the weight matrix of the connection between the hidden layers. Different from RBM, the parameter  $c$  is the offset of hidden nodes. DBM, an undirected multi-layer graph model, combines bottom-up transfer with top-down feedback to get more feature information, and generates samples more reliably dependent on data and high-level features.

**Models Training and Shape Generation.** The goal of training model [15] is to determine the parameters of the model by some given training samples. Contrastive - Divergence algorithm [7] is used to train the RBM deep learning models. The idea of the  $k$ -step Contrastive Divergence learning algorithm (CD- $k$ ) is as follows.

Take the sample in the training set as the initial visible layer of the model, then transfer it to Gibbs [4] for sampling. The initial value is denoted as  $V(0)$ , after  $k$ -step sampling,  $V(k)$  is obtained. In each sampling step, for example step  $t$ , there are two periods: (a) obtain  $h(t)$  from  $p(h|v(t))$ ; (b) then get  $v(t + 1)$  from  $p(v|h(t))$ .

The training of DBN uses the greedy layer-wise unsupervised training strategy [9] divide the process into two phases: pre-training and fine-tuning. Different from the training of DBN, DBM simultaneously synthesizes the information of two adjacent layers. After the models training is completed, the parameters of the models are updated. Then, the images pre-segmented by MRF are sent to the models, the prior shapes of the images are generated by Gibbs sampling [4].

**Shape Prior.** Since the prior shapes and images pre-segmented by MRF are all binary images, shape prior can be obtained from shape similarity measure between them. For the generated prior shape  $\Psi$  and the pre-segmentation image  $f$ , the shape energy term is defined as follows:

$$E^S(f, \psi) = \sum_{p \in P} (H(f_p) - H(\psi_p))^2 = \sum_{p \in P} (H(f_p)\bar{H}(\psi_p) + \bar{H}(f_p)H(\psi_p)) \quad (8)$$

Where  $H(x)$  denotes the step function.  $f$  and  $\Psi$  denote the symbol distance function of the pre-segmentation result and the generated prior shape respectively.  $P$  denotes the set of all pixels in the image, and  $p$  denotes pixel in  $P$ . The value of  $\Psi_p$  (and  $f_p$ ) is either 0 or 1. Then, the global optimal solution of the energy function can be obtained by Graph cuts [10].

### 3 Experimental Results and Analysis

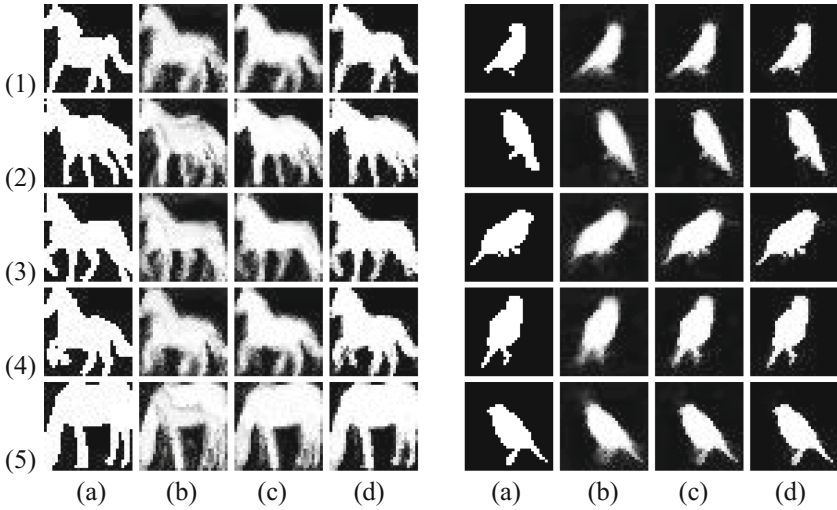
The experimental environment is Matlab R2010b installed under the Win 7 system, and the computer is configured as Intel(R) Xeon(R) CPU E5-2690, 1.9 GHz, 256 GB RAM. Our method is tested on the standard image datasets Weizmann Horse Dataset [13] and CUB\_200\_2011 Dataset [14]. The sizes of images were normalized to 32\*32. The parameters of the model were selected according to the literature [11] and the experiments conducted in this paper.

Figure 4 shows the prior shapes generated by RBM, DBN, and DBM, where Fig. 4(a) is the Ground-truth (artificial segmentation result), Fig. 4(b) is the shape generated by RBM, Fig. 4(c) is the shape from DBN, Fig. 4(d) is the generated by DBM.

It can be observed that deep learning models have strong modeling ability for the objects having the same attributes of the gesture, size, face orientation and shape. In the RBM model modeling results, the edges of the targets are fuzzy and contains many noise points, the details are also missing a lot. The results of DBN and DBM are more complete and clear, DBM generating the best shapes and providing the more effective shape prior for subsequent image segmentation.

Figure 5 shows the segmentation results of some images in the Weizmann Horse Dataset. Comparing our method with the traditional method which does not include the deep learning shape prior under the condition of the same parameter environment and the same initial markers. Figure 5(a) is the original images in Weizmann Horse dataset, Fig. 5(b) is the segmentation results of the traditional MRF method, and Fig. 5(c), (d) and (e) are the MRF image segmentation results combined with RBM, DBN, and DBM shape priors in this algorithm. Figure 5(f) is the result of artificial segmentation.

From the segmentation results that compared with the traditional MRF image segmentation algorithm, our method obtains a complete segmentation results with specific details. The introduction of deep learning shape prior effectively solves the problems. The characteristics of RBM, DBN and DBM reflects in the segmentation results is that introducing the deep learning shape prior can effectively improve the segmentation accuracy.



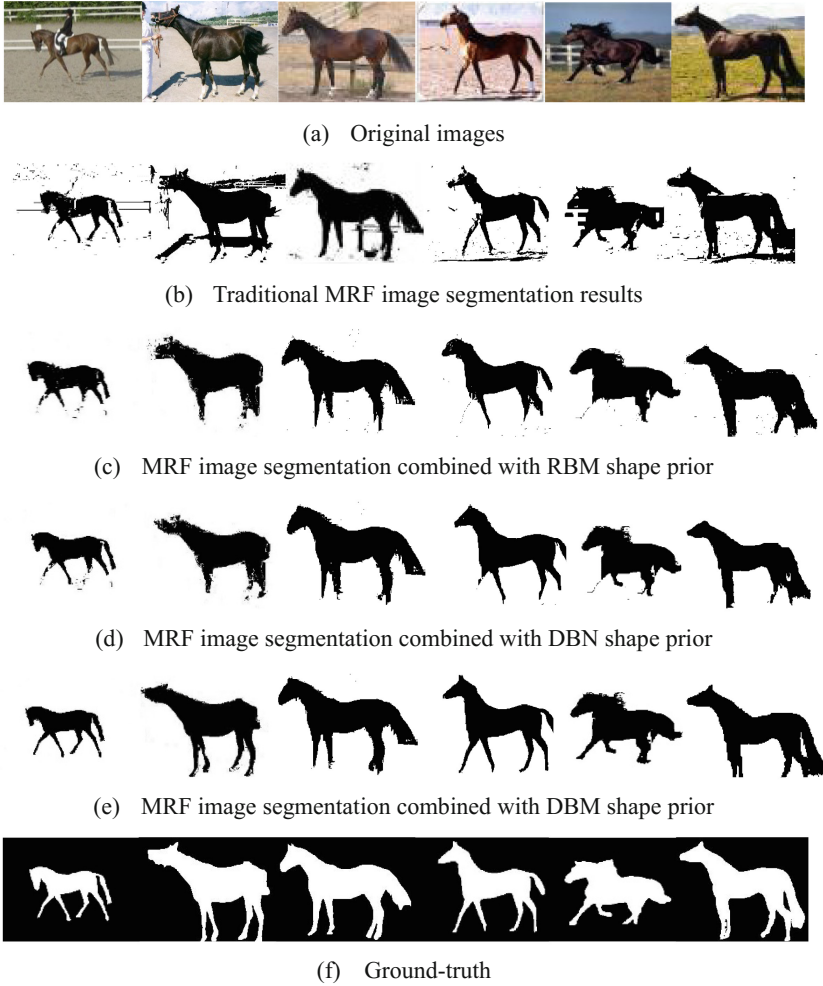
**Fig. 4.** Shapes generated from each model. (a) Ground-truth, (b) RBM, (c) DBN, (d) DBM

Figure 6 shows the bird's original images in the CUB\_200\_2011 Database and the segmentation results of each method. Figure 6(a) is the original images in the dataset, and Fig. 6(b) is the traditional MRF image segmentation results, Fig. 6 (c), (d) and (e) are the results of the MRF image segmentation combined with RBM, DBN, and DBM shape priors in our method. Figure 6(f) is the Ground-truth.

We can see that after using our method, the target can be completely segmented. In the case where the target is in the disorderly branches and is severely obstructed, it is also significantly improved after introducing deep learning shape priors.

Table 1 shows the comparison of the performance of the algorithm using different data sets, including our method and the traditional MRF image segmentation method that does not include shape prior information and the comparison of the method's average segmentation error rate and run time of the data sets. Shape prior makes the time consumption slightly increase, while correct rate of segmentation is effectively improved, and the highest segmentation accuracy rate can reach 99%. The minimum is not less than 90%, indicating that the deep learning shape prior can obtain complete and accurate segmentation results.

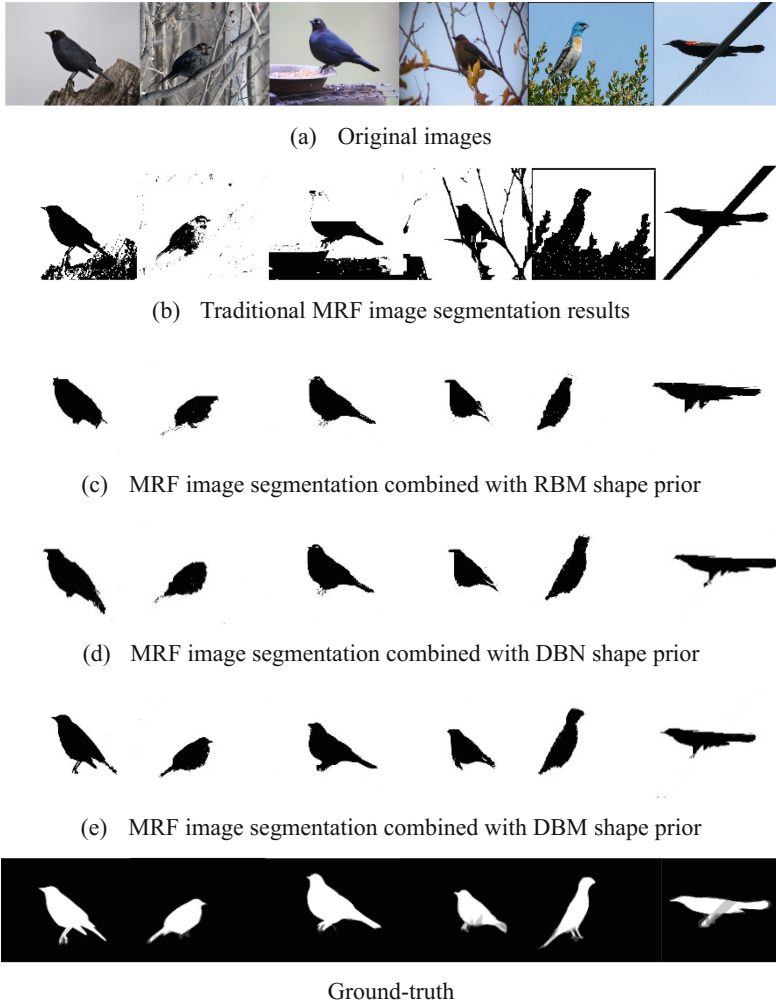
The fact that many researchers have made efforts to introduce shape priors in image segmentation methods, our method is further compared with the existing MRF image segmentation methods with shape priors, based on the Weizmann Horse Database. The methods involved are the MRF based object segmentation combining edge and shape prior and MRF image segmentation method based on shape prior [13, 14], where Fig. 7 is the segmentation and experimental results of only adding shape prior in the other above methods. The data in Table 2 is the segmentation accuracy rate and running time of each method in Fig. 7.



**Fig. 5.** Segmentation results of each method based on Weizmann Horse Dataset

According to the comparison of the experimental data and the result images, it can be concluded that the analysis of quantitative indicators found that the correct rate of segmentation of our method is generally slightly higher than the others, and the running time is also reduced. It can be clearly seen in the image of horse002, the shapes generated by the deep learning models is integrated, making the shape prior fully depict the target shape and guide the segmentation accurately and satisfactorily.





**Fig. 6.** Segmentation results of each method based on CUB\_200\_2011 Dataset

**Table 1.** Comparison of average segmentation accuracy and running time of Weizmann Horse and CUB\_200\_2011 databases

Dataset	MRF	MRF/RBM	MRF/DBN	MRF/DBM
	Accuracy/% time/s	Accuracy/% time/s	Accuracy/% time/s	Accuracy/% time/s
(1)	87.4452 1.3702	95.7566 1.7539	96.4392 1.7723	97.1187 1.7972
(2)	91.0501 1.4861	96.5504 1.8105	98.9471 1.8462	98.4023 1.8672



(a) MRF image segmentation combined with DBM shape prior in our method.



(b) The results of literature [11].



(c) The results of literature [12].

**Fig. 7.** Segmentation results of our method and reference literature methods**Table 2.** Comparison of segmentation accuracy and running time of our method and others methods

Images	Proposed method		Literature [11] method		Literature [12] method	
	Accuracy/%	time/s	Accuracy/%	time/s	Accuracy/%	time/s
Horse 002	98.507	1.819	97.219	1.857	97.178	1.938
Horse 022	98.529	1.786	97.645	1.656	98.240	1.971
Horse 097	98.838	1.894	98.681	1.871	97.737	2.634
Horse 276	98.725	1.724	97.436	1.782	96.968	1.679

## 4 Conclusion

This paper proposes image segmentation based on MRF method combining with deep learning shape priors generated from RBM, DBN, and DBM. The three shape models can learn shape features automatically, express shape flexibly, even if the input objects and the training samples are different in orientation, size and posture. For the input images that do not appear in training dataset and contain more interference factors, deep learning models still can reconstruct the shape ideally. Better segmentation results can be obtained after introducing the generated shape priors to the MRF-based image segmentation framework. The experimental results of Weizmann Horse Dataset and CUB\_200\_2011 Dataset indicate the impact of shape prior especially under the

complex situation of similar target and background, noisy image, object occlusion, background clutter, etc. The generated shape prior can effectively constrain the target, so as to obtain more complete and accurate segmentation result. Compared with the traditional MRF image segmentation method, the proposed method can get better segmentation results.

## References

1. Belongie, S., Carson, C., Greenspan, H., et al.: Color-and texture-based image segmentation using EM and its application to content-based image retrieval. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 675–682 (1998)
2. Chen, J., Pappas, T.N., Mojsilovic, A., et al.: Adaptive image segmentation based on color and texture. In: Proceedings of the IEEE International Conference on Image Processing, pp. 777–780 (2002)
3. LeCun, Y., Bengio, Y., Hinton, G.E.: Deep learning. *Nature* **521**(7553), 436–444 (2015)
4. Juan, Z., Xili, W., Jianguo, Y.: Shape modeling method based on deep learning. *Chin. J. Comput.* **41**(01), 132–144 (2018)
5. Hinton, G.E., Osindero, S., Teh, Y.-W.: A fast learning algorithm for deep belief nets. *Neural Comput.* **18**(7), 1527–1554 (2006)
6. Salakhutdinov, R., Hinton, G.E.: Deep Boltzmann machines. *J. Mach. Learn. Res.* **5**(2), 1967–2006 (2009)
7. Hinton, G.E.: Training products of experts by minimizing contrastive divergence. *Neural Comput.* **14**(8), 1771–1800 (2002)
8. Huang, L., Nie, J., Wei, Z.: Human body segmentation based on shape constraint. *Mach. Vis. Appl.* **2017**(2), 1–10 (2017)
9. Hinton, G.E.: A practical guide to training restricted Boltzmann machines. *Momentum* **9**(1), 599–619 (2010)
10. Boykov, Y., Kolmogorov, V.: An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Trans. Pattern Anal. Mach. Intell.* **26**(9), 1124–1137 (2004)
11. Zhang, W.: Markov random field based object segmentation combining edge and shape prior. *J. Chongqing Univ. Technol.* **10**, 79–85 (2014)
12. Zhang, W.: Research on Image Classification Based on Probability Graph Model. ShaanXi Normal University, Xi'an (2013)
13. Borenstein, E., Sharon, E., Ullman, S.: Combining top-down and bottom-up segmentation. In: CVPR (2004)
14. Welinder, P., Branson, S., Mita, T., Wah, C., Schroff, F., Belongie, S., Perona, P.: Caltech-UCSD Birds 200. Technical report CNS-TR-2010-001, California Institute of Technology (2010)
15. Fischer, A., Igel, C.: Training restricted Boltzmann machines: an introduction. *Patt. Recogn.* **47**(1), 25–39 (2014)
16. Elfwing, S., Uchibe, E., Doya, K.: Expected energy-based restricted Boltzmann machine for classification. *Neural Netw.* **64**, 29–38 (2015)



# An Automated Matrix Profile for Mining Consecutive Repeats in Time Series

Mahtab Mirmomeni<sup>1</sup>(✉), Yousef Kowsar<sup>1,2</sup>, Lars Kulik<sup>1</sup>, and James Bailey<sup>1</sup>

<sup>1</sup> The University of Melbourne, Melbourne, Australia  
{m.mirmomeni,y.kowsar}@student.unimelb.edu.au,  
{lkulik,baileyj}@unimelb.edu.au

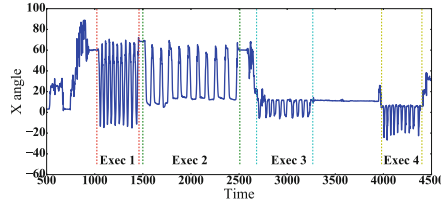
<sup>2</sup> Microsoft Research Centre for Social NUI, Melbourne, Australia

**Abstract.** A key application of wearable sensors is remote patient monitoring, which facilitates clinicians to observe patients non-invasively, by examining the time series of sensor readings. For analysis of such time series, a recently proposed technique is Matrix Profile (MP). While being effective for certain time series mining tasks, MP depends on a key input parameter, the length of subsequences for which to search. We demonstrate that MP's dependency on this input parameter impacts its effectiveness for finding patterns of interest. We focus on finding consecutive repeating patterns (CRPs), which represent human activities and exercises whilst tracked using wearable sensors. We demonstrate that MP cannot detect CRPs effectively and extend it by adding a locality preserving index. Our method automates the use of MP, and reduces the need for data labeling by experts. We demonstrate our algorithm's effectiveness in detecting regions of CRPs through a number of real and synthetic datasets.

## 1 Introduction

Activity and exercise detection using wearable sensors helps clinicians to remotely monitor and better diagnose patients' movements non-invasively [1]. A key task is to find patterns of interest in the time series data generated by wearable sensors. These patterns can be indicative of the patient's status, for example, showing the manner in which a patient is performing a rehabilitation exercise.

Recently, a technique known as Matrix Profile (MP) has been proposed to mine time series data. Despite MP's advantages, exactness, space efficiency and tolerance to missing data, it faces two fundamental challenges: its sensitivity to a key input parameter and its inability to detect CRPs. The authors of the MP assume that the method is effectively parameter-free: "In contrast, our proposed algorithm has zero parameters to set" [2] and the input parameter is based on "user choice" [3], and "our algorithm is insensitive to the value of the only input parameter" [4]. In Sect. 2, we demonstrate that, MP is highly sensitive to its input parameter, the length of subsequence used for searching and MP is limited in detecting CRPs.



**Fig. 1.** A time series epoch captured from post-rehabilitation exercises, where 4 different exercises (CRPs) have been performed by a particular patient.

Figure 1 shows an example of a time series interval (known as epoch) recorded by an accelerometer and collected from a patient performing rehabilitation exercises, moving their leg relative to the ground whilst wearing an ankle cuff with an embedded accelerometer. The repeating patterns correspond to exercises being repetitively performed. Regions between the exercises indicate other activities or breaks between the exercises.

For MP to be able to detect the rehabilitation exercises in Fig. 1, it is not reasonable to expect the user (e.g. a physiotherapist) to set the input parameter for MP. Rather, it must be done automatically. To this end, we develop a technique to automatically select the subsequence length for MP that can accurately identify the CRPs. To overcome the MP’s limitation in detecting CRPs, we extend MP by adding a new index that preserves the locality of the repeats.

We provide a theoretical justification for how our new index can be used to detect the regions of CRPs from a given time series. We show that using our method, we can automatically set MP’s input parameter, so that it can accurately identify CRPs in a number of synthetic and real datasets.

## 2 Related Work and Limitations

**Activity and Exercise Detection Background.** Activity recognition using wearable sensors has gained a lot of popularity given their rise in the consumer space [1,5,6]. A common approach to detect activity and exercises is to use supervised or semi-supervised methods, such as statistical, hidden Markov or mixture models [6–8]. The supervised and semi-supervised methods, however, need domain expertise to label the data, which makes their use in real world applications limited.

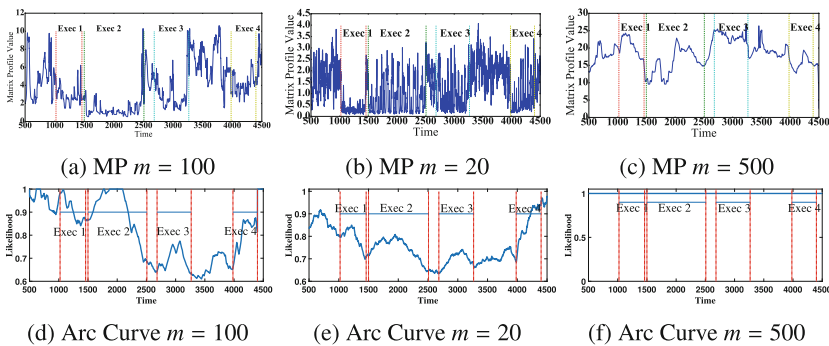
Another common approach to detect activity and exercises is through motif discovery techniques on time series data, which are of an unsupervised nature [4, 9,10]. Motifs are previously unknown subsequences that have been repeated over time [11]. Our problem of finding CRPs in a time series is different to motif discovery algorithms: we aim to find a burst of repeating patterns that happen at a specific point in the time series, e.g., when a patient performs an exercise.

The Matrix Profile (MP) is the newest technique in discovering similar patterns in a time series [2]. We explore the use of the MP for detecting CRPs, corresponding to an exercise in the time series generated by a wearable sensor.

**Matrix Profile Background.** Matrix profile (MP) [2] is the most recent technique for all-pair-similarity-search within a time series. A *Time Series* is a sequence of real value numbers observed in time  $T = \{t_1, t_2, \dots, t_n\}$ , where  $n$  is the length of  $T$ . A *subsequence* of a given time series  $T$  is a time series starting from  $t_i$  with length  $m < n + 1 - i$ . In an all-pair-similarity-search, the distances between every subsequence in a time series with all subsequences are calculated. By definition MP is a vector that for each subsequence of a specified length ( $m$ ) in the time series, stores the smallest Euclidean distance between that subsequence and its nearest neighbour in the time series [2]. The index of the nearest neighbour is stored in another vector called the Index Profile (IP) [2].

**Sensitivity to Key Input Parameter:** Figure 2a shows the MP that corresponds to the time series in Fig. 1, with input parameter  $m = 100$ . In the exercise regions, marked on the Fig. 2a, the MP’s values drop to a local minimum (often close to zero). Figures 2b and c show MP for the same time series in Fig. 1 with  $m = 20$  and  $m = 500$ . Comparing the behaviour of MP in these 3 figures (Fig. 2a, b and c), we see that MP significantly depends on the value of subsequence length  $m$ .  $m$  can be seen as the granularity level for subsequence searching in a time series. Setting  $m$  too large (in our example 500) results in comparing long subsequences from the time series that reduces the chance of finding similar subsequences (Fig. 2c). Setting  $m$  too small (20 in our example) results in most of the subsequences being assessed similar with each other, which can show itself as sudden fluctuations in the resulting MP (Fig. 2b).

The closest application of MP to our problem is the semantic segmentation of a time series [4]. The authors introduced Arc Curves, a transformation of the time series into a new plot that at each point annotates the time series with likelihood of regime change, using IP. We investigated the Arc Curve algorithm [4] for detecting exercise regions of CRPs and applied the code provided by the authors to our rehabilitation exercise dataset. Figure 2(d, e, f) shows the Arc Curves for the time series in Fig. 1 using different subsequence lengths, and demonstrates



**Fig. 2.** MP (Top row) and Arc Curve (Bottom row) with various input parameter  $m$  for time series in Fig. 1. Both MP and Arc Curve values are heavily dependant on the value of the input parameter  $m$ , length of subsequence searching.

that Arc curves are also highly sensitive to changes in the input parameter. Despite our best efforts, we were not able to produce segments corresponding to regions of CRPs, using the provided code.

**MP’s Limitations for Detecting CRPs:** We define *Region of CRPs* to be the region where the same pattern is consecutively repeated. Let *Region of CRP*,  $RS$ , to be the region that  $\exists f$  (function of repeat),  $\Delta t > 0 | \forall x \in RS, f(x) = f(x + \Delta t)$ . We call the pattern that is repeating consecutively inside a  $RS$ , the Signal of Repeat. As shown in Fig. 2a, MP value drops close to zero when detecting a repeating pattern. However, the repeating patterns do not necessarily need to be consecutive for this to happen. Figure 3 shows a time series with non-CRPs and the corresponding MP. The value of MP is close to zero for a repeating pattern, although the patterns are not CRPs. To solve this problem, we need to determine whether the most similar patterns are also temporally close together. We next outline how to define an index that preserves the locality of the repeats.

### 3 Problem Statement

To overcome MP’s limitation of preserving the locality of repeating patterns, we define Distance Index (DI) as a vector that at each point stores the distance between the index of any subsequence of length  $m$  of a time series to the index of its nearest neighbour. We formally define DI as follows:

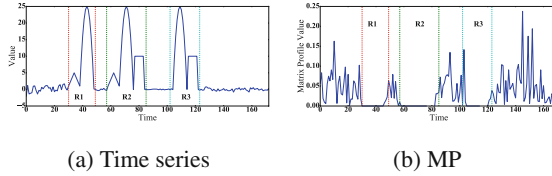
**Definition 1.**  $\{Distance\ Index\}$  is a vector of distances, where  $DI[i] = i - Index\ Profile[i]$

Figure 4a shows the corresponding DI with  $m = 100$  of the time series in Fig. 1 and MP in Fig. 2a. In Sect. 3, we show that the value of DI in the repeat section is equal to the period of the repeating pattern.

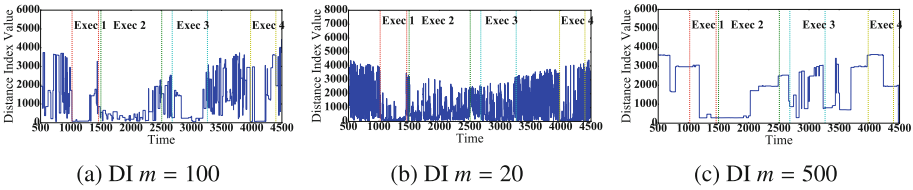
**The Most Repeat-Sensitive DI for Detecting CRPs.** In Sect. 2, we showed that MP is sensitive to its key input parameter  $m$ . Thus finding the right value for the subsequence length  $m$  is crucial for using MP in different applications. To find this value, we need to determine which input parameter  $m$  results in a DI that best detects the area of CRPs. We define the most repeat-sensitive DI as a DI that aligns to the period of the repeating pattern and stays flat for the duration of the repeat.

**Definition 2.**  $\{Most\ Repeat-sensitive\ Distance\ Index(DI)\}$  The most repeat-sensitive Distance Index for finding region of CRPs,  $RS$ , is a DI, such that  $\forall x \in RS: DI = \Delta t$ .

We can show that setting  $m$  to the smallest subsequence that is not repeating within the signal of repeat, results in a DI with a flat region equal to the repeat period, in the regions of CRPs.



**Fig. 3.** Motivation example for extending MP with Distance Index: Time series with repeating patterns that are not consecutive with the corresponding MP, which still drops to zero at repeats.



**Fig. 4.** Distance Index (DI) with various input parameter  $m$  for time series in Fig. 1. In the repeating regions, DI stays flat at the value of the repeat period for the duration of the repeat period. Although susceptible, DI is more robust to changes in  $m$ .

**Lemma 1.** Given a time-series  $T$  with region of CRPs,  $RS$ , function of repeat  $f(t)$  and period of repeat  $\Delta t$ ,  $RS = \{t_i | f(t_i + \Delta t) = f(t_i)\}$ . A MP with  $m = \Delta t \rightarrow DI = C$  in the corresponding region of CRPs, where  $\forall t_i \in RS | C = \Delta t$ .

*Proof.* The proof results from the definition of CRPs region,  $RS$ , that matches the periodic function definition. For any subsequence inside  $RS$  (except the first and last subsequences) the closest repeating pattern is one period away from the subsequence. The DI values for the first and last subsequences depend on the preceding and succeeding subsequences of  $RS$ , which are not part of the CRPs.

**Lemma 2.** Given a Region of Repeat,  $RS$ , with period  $\Delta t$  and signal of repeat  $S$ , the DI value for any subsequence within the signal of repeat is equal to  $\Delta t$  iff the subsequence is not repeating within the signal of repeat.

A direct conclusion from Lemmas 1, 2 is that the input parameter  $m$  (subsequence length) for finding the most repeat-sensitive DI is bounded by the period of the region of repeat. We include these findings in the following theorem:

**Theorem 1.** Given a time-series  $T$  and a region of CRPs,  $RS$ , with period  $\Delta t$ , the best value of  $m$  for finding the region of repeat is equal to the length of the shortest subsequence that is not repeating within the signal of repeat.



**Automatically Determining the Best  $M$ .** In an unsupervised scenario, where the ground truth is not known, we need to define a mechanism to find the most-repeat sensitive DI from a pool of DIs calculated using a range of subsequence length  $ms$ . In the regions of CRPs the value of MP must be close to zero, and the area under MP can be used to find the most repeat-sensitive DI. Thus, for each DI, we find the flat segments and nominate them as regions of CRPs. We calculate the area under the corresponding MPs for those regions and select the DI corresponding to the MP with the minimum area as the most repeat-sensitive DI. Theorem 1 shows that the upper bound value for  $m$  to produce a repeat sensitive DI is equal to the period for that region. Thus a brute force search to find best  $m$  is of  $O(n^2 \times \text{estimated period})$  in time. We use the inverse of the most dominant frequency from time series' Fourier transformation to estimate the period.

## 4 Experimental Evaluation

The purpose of our experiments<sup>1</sup> is to evaluate how accurately we can identify the CRP regions of a time series (the regions of exercises in our physiotherapy dataset) using MP, when automatically setting the value for input parameter  $m$ , length of subsequence searching. To evaluate our algorithm, we use ground truth provided by experts on the location of exercises in an epoch.

We calculate DI using a range of subsequence length  $ms$  and find flat segments of each DI. We set the value of all points on the flat segment of DI to 1 and the rest to 0. For the ground truth, we set the value of all points in the repeating region  $RS$  to 1 and the rest to 0. We define *true positive* as the number of points with value 1 on DI that align with the region corresponding  $RS$  and *false positive* as the number of points on DI that have value 1, but are outside of  $RS$ . For each DI, we report the F1-Score and the Adjusted Mutual Information (AMI) between DI with the ground truth using the True/False positives.

We created two synthetic datasets. The first synthetic dataset is a simple sine waveform with a period of 180. The sine waveforms are preceded and succeeded by Gaussian noise. The second synthetic dataset contains repeats within repeats. The repeating sequence is a waveform with a period of 30, which contains two repetitive waveforms with a period of 10.

We used a real Physiotherapy dataset, collected by the Physiotherapy department at the University of Melbourne, of patients with chronic knee pain performing 4 rehabilitation exercises, while wearing an ankle-cuff with an embedded accelerometer [12]. We used data from 10 patients. The accelerometer's angle with respect to the  $x$  axis is of interest, in this context. To detect regions of CRPs in an epoch, we use a tumbling window of size 1000 to segment the epoch and search within that tumbling window. We perform our search for  $m$  in the range of  $m \in [2, \text{period}]$ , according to the upper limit set for  $m$  described in Sect. 3. The brute-force algorithm searches for a best  $m$  in the range of  $m \in [2, \text{window length}/3 \approx 300]$ . The cut off value for the brute-force search is set so that we have at least three CRPs in our window.

<sup>1</sup> Our code is available on <http://goo.gl/TLfCLp>.

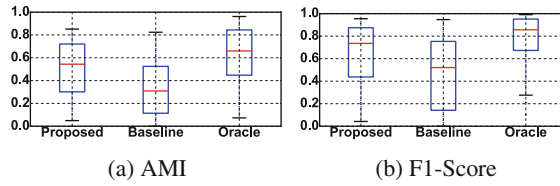
**Results and Discussion.** We evaluate how accurately we can find the region of CRPs using the proposed input parameter  $m$ . Our synthetic data has a fixed period of repeats. According to Theorem 1 the best value of  $m$  for finding region of repeats is equal to the length of the shortest subsequence that is not repeating within the signal of repeat. For the first Synthetic dataset, subsequence length  $m$  was found to be equal to 4 using both our proposed algorithm and the algorithm that searches for the best  $m$ , which results in AMI and F1-Score of 0.99. For the second synthetic dataset, subsequence length  $m$  was found to be 11, where 10 is the length of the subsequence repeating in the repeat signal, using both algorithms. The subsequence of length 11 results in AMI and F1-Score of 0.98. Both results agree with Theorem 1.

For our real dataset, since each repeat of an exercise varies slightly from its surrounding repeats, the period for the region of CRPs is unknown and can only be estimated. As a result, instead of observing one flat line in DI, corresponding to a region of repeat, we observe discontinuity in the flat line corresponding to the region of repeat. We estimate the period by taking the mean from the discontinued flat line, which in turn causes  $m$  found by our proposed algorithm to be different to the best  $m$ , found using a supervised approach. It is not possible to stipulate a single, universal value for  $m$ .

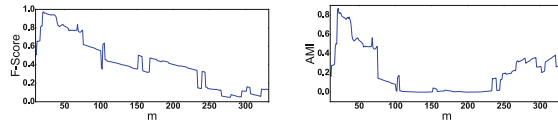
To set a baseline, we used a randomly generated  $m$  based on domain knowledge. According to experts, each exercise set in our Physiotherapy dataset on average takes between 30s–1 min. There are 10 repeats on each exercises set, therefore each repeat takes 3–6s (on average). Data sampling rate is 12 data points per second, therefore each region’s period is expected to be between 36 ( $3 * 12$ ) to 72 ( $6 * 12$ ) points. We set  $m$  to a random number between 36 to 72 for each epoch as the baseline.

Figure 5a and b depict AMI (median = 55%) and F1-Score (median = 74%) of our proposed algorithm for finding CRPs vs the AMI (median = 61%) and F1-Score (median = 38%) for the baseline algorithm. In both box-plots we observe that our proposed method for selecting  $m$  results in a more accurate detection of region of CRPs. In both cases our proposed method significantly outperforms the baseline (AMI paired t-test p-value  $< 0.0002$  and F1-score paired t-test p-value  $< 0.0001$ ). In both diagrams, the baseline method has a wider range of values. This results from the sensitivity of MP to the input parameter  $m$  and that setting  $m$  based on domain knowledge cannot overcome this sensitivity. We have included the best  $m$  results as an oracle (AMI median = 66%, F1-Score median = 86%) for comparison. Finding the best  $m$  results requires labeled data. Since we are using an unsupervised approach and we want our method not to be dependant on labelled data, the best  $m$  is unknown.

We investigate the effect of changes in  $m$  on F1-Score and AMI for finding CRPs in Fig. 6. In these plots, we are evaluating Theorem 1 for a randomly selected window from our Physiotherapy dataset. From both plots, it is evident that setting  $m$  too small results in a poor detection of regions of CRPs. However, the accuracy of the proposed method surges in both evaluation metrics as input parameter  $m$  increases and drops as input parameter  $m$  gets too large, which fully agrees with Theorem 1.



**Fig. 5.** Boxplot of comparison for finding region of CRPs using our proposed input parameter vs domain-knowledge (Baseline) and Oracle  $m$  for the Physiotherapy dataset.



**Fig. 6.** F1-Score (left) and AMI (right) changes over changes in input parameter  $m$ .

## 5 Conclusion

We explore the use of the MP to detect CRPs, that translate to exercises in our Physiotherapy dataset. We show that MP has two fundamental limitations: it is sensitive to a key input parameter, the subsequences length for which to search, and does not detect if repeating patterns are consecutive. We introduce a new index, DI, to preserve the locality of repeats. The most repeat-sensitive DI can accurately identify the region of CRPs. We prove that to achieve the most repeat-sensitive DI, the input parameter  $m$  has to be set to the length of the shortest subsequence that is not repeating within the signal of repeat. We compare the accuracy of our unsupervised algorithm in finding the CRP regions to the results from applying the best  $m$ , calculated using a supervised method. The comparison shows that we can, with high accuracy (Average difference AMI 12%), automatically and without a priori knowledge about the regions, find the regions of CRPs (exercises in our Physiotherapy dataset). Our proposed method finds the input parameter  $m$  that outperforms selecting  $m$  using domain knowledge by 15%.

## References

1. Andreu-Perez, J., Leff, D.R., Ip, H.M.D., Yang, G.Z.: From wearable sensors to smart implants toward pervasive and personalized healthcare. *IEEE Trans. Biomed. Eng.* **62**(12), 2750–2762 (2015)
2. Yeh, C.-C.M., Zhu, Y., Ulanova, L., Begum, N., Ding, Y., Dau, H.A., Silva, D.F., Mueen, A., Keogh, E.: Matrix profile I: all pairs similarity joins for time series: a unifying view that includes motifs, discords and shapelets. In: *Proceedings of ICDM* (2016)

3. Zhu, Y., Zimmerman, Z., Senobari, N.S., Yeh, C.C.M., Funning, G., Mueen, A., Brisk, P., Keogh, E.: Matrix profile II: exploiting a novel algorithm and GPUs to break the one hundred million barrier for time series motifs and joins. In: Proceedings of ICDM (2016)
4. Gharghabi, S., Ding, Y., Yeh, C.C.M., Kamgar, K., Ulanova, L., Keogh, E.: Matrix profile VIII: domain agnostic online semantic segmentation at superhuman performance levels. In: Proceedings of ICDM (2017)
5. Patel, S., Park, H., Bonato, P., Chan, L., Rodgers, M.: A review of wearable sensors and systems with application in rehabilitation. *J. Neuro Eng. Rehabil.* **9**(1), 21 (2012)
6. Kwapisz, J.R., Weiss, G.M., Moore, S.A.: Activity recognition using cell phone accelerometers. *SIGKDD Explor. Newsl.* **12**(2), 74–82 (2011)
7. Kowsar, Y., Moshtaghi, M., Velloso, E., Kulik, L., Leckie, C.: Detecting unseen anomalies in weight training exercises. In: Proceedings of OzCHI (2016)
8. Minnen, D., Isbell, C.L., Essa, I., Starner, T.: Discovering multivariate motifs using subsequence density estimation and greedy mixture learning. In: Proceedings of the 22nd National Conference on Artificial Intelligence, vol. 1, pp. 615–620. AAAI Press (2007)
9. Minnen, D., Starner, T., Essa, I., Isbell, C.: Improving activity discovery with automatic neighborhood estimation. In: Proceedings of IJCAI, pp. 2814–2819 (2017)
10. Vahdatpour, A., Amini, N., Sarrafzadeh, M.: Toward unsupervised activity discovery using multi-dimensional motif detection in time series. In: Proceedings of IJCAI, pp. 1261–1266 (2009)
11. Chiu, B., Keogh, E., Lonardi, S.: Probabilistic discovery of time series motifs. In: Proceedings of SIGKDD, pp. 493–498 (2003)
12. Bennell, K.: Adherence to home exercises in the treatment of knee osteoarthritis. <https://healthsciences.unimelb.edu.au/research-groups/physiotherapy-research/chesm/more>



# High-Resolution Depth Refinement by Photometric and Multi-shading Constraints

Yujun Zhang<sup>1,2</sup>, Qian Zhang<sup>1,2</sup>, and Wei Feng<sup>1,2</sup>(✉)

<sup>1</sup> School of Computer Science and Technology, Tianjin University, Tianjin, China  
wfeng@tju.edu.cn

<sup>2</sup> Key Research Center for Surface Monitoring and Analysis of Cultural Relics, SACH, Beijing, China

**Abstract.** Depth refinement is critical for consumer depth cameras to remove the inherent noises and unreliable or missing data. In this paper, we propose a simple yet effective approach to achieve high-resolution refinement of low-cost Kinect depth map. The key of our approach is a unified energy function with two new constraints terms, that is, the photometric and multi-shading gradients constraints. Specifically, photometric constraint term helps to enrich the faithful local details of scene surface; while multi-shading gradients constraint term suppresses the effect of inaccurate normal estimation under multiple illuminations. Besides, smoothness and initial depth constraints are also included. We design an adaptive weighting strategy to further increase the robustness of approach for the depth-missing and non-Lambertian regions. Since energy function can be optimized directly in the shading domain, the refined depth map has the same higher resolution of the shading images. Experiments validate the effectiveness of our approach with reliable accuracy and faithfully richer 3D details than competitor methods.

**Keywords:** Depth refinement · Consumer camera  
High-resolution 3D reconstruction · Photometric stereo  
Multi-shading gradients

## 1 Introduction

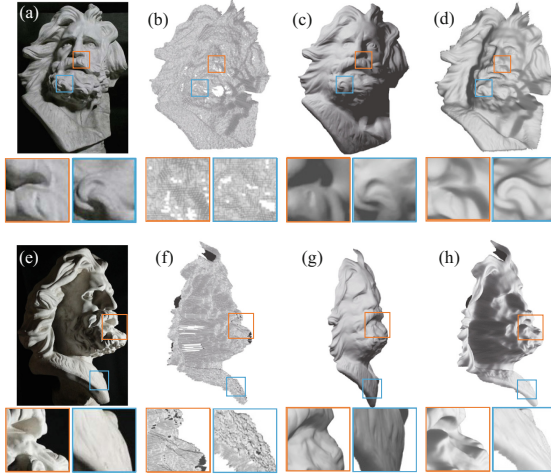
Since consumer depth cameras can provide both RGB images and depth map in a low-cost way, it brings many opportunities for a lot of challenging problems in computer vision and graphics [9]. Based on our existing work, good quality depth map can help enhance the accuracy of active lighting recurrence [18], apply for fine-grained change detection in 3D structure [5], and improve the efficiency of active camera relocalization [4, 11, 15]. As shown in Fig. 1, the depth map

---

Y. Zhang and Q. Zhang—Contributed equally to this work.

W. Feng—This work is supported by NSFC 61671325, 61572354, 61672376.

captured by a depth camera is globally accurate but it contains excessive noise and depth-missing areas. In contrast, photometric stereo (PS) [16] can generate good quality local surface details, but fails to preserve global structure.



**Fig. 1.** Illustration of the advantage of combining consumer camera captured depth map and photometric stereo (PS). (a) A typical RGB image captured by Kinect II. (b) 3D reconstruction result from depth map. (c) 3D reconstruction via a state-of-the-art uncalibrated PS method [13]. (d) 3D reconstruction by our the proposed approach. (e)–(h) The side view results of (a)–(d), respectively.

So far, a typical way is to combine shape from shading (SfS) [8]. [6, 10, 12, 17] try to use single image shading constraint to refine depth map. Although those SfS-based methods are simpler and faster, they always need prior knowledge to solve the ill-posed problem and the refinement accuracy cannot be guaranteed. In addition, photometric stereo (PS) [16] focuses on acquiring surface shape and reflectance from multi-illumination images. [1, 3, 7, 14] try to use PS to obtain detailed local structure for depth refinement. Besides, depth refinement by fusing multi-view stereo has also been tried many times [2, 19, 20].

In this paper, we propose a feasible depth refinement approach based on the essential spirit of photometric stereo, by combining multi-shading gradients constraint under multiple illuminations and the photometric constraint. To be best of our knowledge, this is the first work using multi-shading gradients constraint. Besides, a smoothness constraint scene and an initial depth constraint are also considered in our model. Both of them can restrain depth noise and fill up the depth-missing area. An adaptive weighting strategy is utilized to increase the robustness of our model. Besides, since we can effectively optimize our model via Gauss-Newton method in the shading domain, our approach can generate

higher-quality higher-resolution depth map ( $1920 \times 1080$ ) than the initial depth ( $512 \times 424$ ). Our major contribution is three-fold:

1. We present an effective unified energy combing both multi-shading gradients constraint and photometric constraint for depth refinement.
2. We design an adaptive weighting strategy to increase the robustness of our approach for depth-missing and non-Lambertian areas.
3. Our method can generate a high-resolution refined depth map from the low-resolution consumer camera, like Kinect II, captured depth maps.

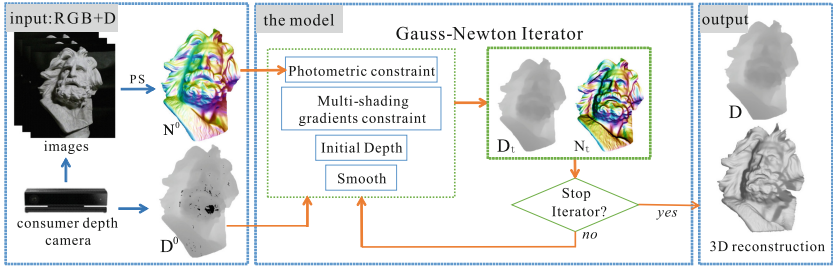


Fig. 2. Framework of the proposed method. See text for details.

## 2 Our Approach

Given RGB images under various illuminations, we employ an uncalibrated PS algorithm to estimate normal and lighting information. In addition, we preclude inaccuracies of sketchy shading model by gradient-based shading under multi-illumination constraints. A significant advantage of this method is that we can not only get strong absolute depth accuracy, but also estimate fine surface details (Fig. 2).

### 2.1 The Model

First, we capture  $K$  images under multiple illuminations, then we use an uncalibrated PS method [13] to calculate the scene normal  $\mathbf{N}^0 \in \mathbb{R}^{P \times 3}$  and lighting direction  $\mathbf{L}_k \in \mathbb{R}^3$  corresponding to  $k$ -th image. To obtain the refined depth map, we minimize the following cost function

$$E(\mathbf{D}) = \omega_g E_g(\mathbf{D}) + \omega_n E_n(\mathbf{D}) + \omega_d E_d(\mathbf{D}) + \omega_s E_s(\mathbf{D}), \quad (1)$$

where  $E_g$  is the multi-shading gradients constraint term,  $E_n$  is the photometric constraint term,  $E_d$  is the depth term, and  $E_s$  is the smoothness term.

**Multi-shading Gradients Constraint Term.** In order to correct the inaccurate depth map caused by non-Lambertian region, e.g., specularity or cast shadow, we penalize the differences between rendered shading gradients and intensity image gradients for the adjacent pixels. We have

$$E_g(\mathbf{D}) = \sum_k^K \sum_p^P \sum_{q \in \mathcal{L}(p)} [(\mathbf{S}_{kp} - \mathbf{S}_{kq}) - (\mathbf{I}_{kp} - \mathbf{I}_{kq})]^2, \quad (2)$$

where  $p$  or  $q$  denotes pixel index,  $k$  is image index,  $\mathcal{L}(p)$  indicates the two neighbors (right, down) of the  $p$ -th pixel,  $\mathbf{S}_k$  and  $\mathbf{I}_k \in \mathbb{R}^P$  indicate  $k$ -th rendered shading and intensity images, respectively. According to the Lambertian lighting model, we have  $\mathbf{S}_k = \mathbf{N}\mathbf{L}_k$ . We directly obtain lighting direction  $\mathbf{L}_k$  from PS [13] method,  $\mathbf{N} \in \mathbb{R}^{P \times 3}$  is normal estimated from depth map  $\mathbf{D}$ . Refer to Sect. 2.3 for the details about the transformation between  $\mathbf{N}$  and  $\mathbf{D}$ .

**Photometric Constraint Term.** The depth map captured by consumer depth cameras is globally accurate but it contains excessive noises and depth-missing areas. PS method can generate good quality local surface details, but fails to preserve global structure. Combining the advantages of PS, we have

$$E_n(\mathbf{D}) = \sum_p^P \|\mathbf{N}_p - \mathbf{N}_p^0\|_2^2, \quad (3)$$

where  $\mathbf{N}_p \in \mathbb{R}^3$  denotes the normal corresponding to  $p$ -th pixel,  $\mathbf{N}^0$  indicates the normal map acquired by the state-of-the-art PS method [13].

**Depth Term.** In order to enforce the refined depth stay close to the valid initial depth  $\mathbf{D}_p^0$ ,  $\mathbf{D}_p \in \mathbb{R}$  stands for the depth value of the  $p$ -th pixel. It satisfies

$$E_d(\mathbf{D}) = \sum_p^P (\mathbf{D}_p - \mathbf{D}_p^0)^2, \quad (4)$$

**Smoothness Term.** Besides, we also employ a smoothness constraint

$$E_s(\mathbf{D}) = \sum_p^P \left\| \mathbf{X}_p - \alpha \sum_{q \in \mathcal{N}(p)} \mathbf{X}_q \right\|_2^2, \quad (5)$$

where  $\mathbf{X}_p$  and  $\mathbf{X}_q \in \mathbb{R}^3$  indicate  $p$ -th and  $q$ -th 3D positions in camera coordinate system,  $\mathcal{N}(p)$  is the set containing the coordinates of the four neighbors around  $p$ -th pixel. We let weights  $\alpha$  be 0.25 in our all experiments. Refer to Sect. 2.3 for details about the transformation between  $\mathbf{X}$  and  $\mathbf{D}$ .



## 2.2 Optimization

Refer to Eq. (1), we estimate the optimal depth map  $\widehat{\mathbf{D}}$  by the following minimization problem

$$\widehat{\mathbf{D}} = \arg \min_{\mathbf{D}} E(\mathbf{D}). \quad (6)$$

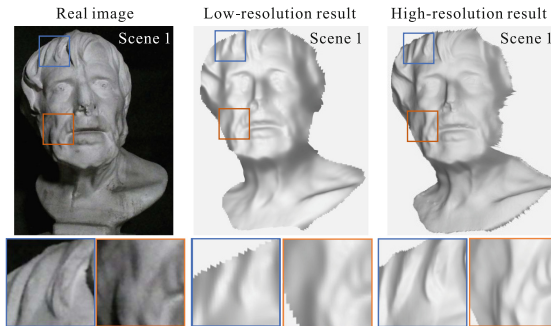
The total number of square terms ( $M = 9P$ ) consists of multi-shading gradients constraint term ( $2P$ ), photometric constraint term ( $3P$ ), depth term ( $P$ ) and smoothness term ( $3P$ ). Equation (6) can be rewritten as

$$E(\mathbf{D}) = \sum_{r=1}^M F_r(\mathbf{D})^2, \quad (7)$$

where  $F_r(\mathbf{D})^2$  indicates  $r$ -th square term. We employ Gauss-Newton method to solve the above non-linear least square problem.

## 2.3 The Algorithm and Implementation Details

Depth map represents the depth (range) between camera and surface point, therefore, we can obtain the 3D position in camera coordinate system of the surface point with camera intrinsic parameter. We have  $\mathbf{X}(i, j) = [\frac{i-u_x}{f_x}, \frac{j-u_y}{f_y}, 1]^T \mathbf{D}(i, j)$ , where  $i$  and  $j$  indicate the row and column coordinates,  $\mathbf{X}(i, j) \in \mathbb{R}^3$  stands for 3D position coordinate of the  $(i, j)$ -th pixel in camera coordinate system,  $[f_x, f_y]$  and  $[u_x, u_y]$  are the known camera focal length and principal point, respectively. Consequently, we can obtain unnormalized surface normal  $\tilde{\mathbf{N}}(i, j) \in \mathbb{R}^3$  at  $(i, j)$ -th pixel combined with neighboring depth pixels by  $\tilde{\mathbf{N}}(i, j) = (\mathbf{X}(i, j-1) - \mathbf{X}(i, j)) \times (\mathbf{X}(i-1, j) - \mathbf{X}(i, j))$ . Then, we can calculate the normalized surface normal  $\mathbf{N}$  from  $\tilde{\mathbf{N}}$  easily.



**Fig. 3.** Our method generate a higher resolution refined result for depth camera. The first column is a real RGB image, the second and third columns denote the 3D reconstruction results of initial depth and our refined depth map.

In addition, considering depth map is less reliable in discontinuity area, we should weaken initial depth influence correspondingly. Similarly, in order to reduce the texture-copy artifacts, the RGB boundary edges should have a lower influence in multi-shading gradients constraint term. So we design an adaptive weighting strategy to further increase the robustness of our approach.

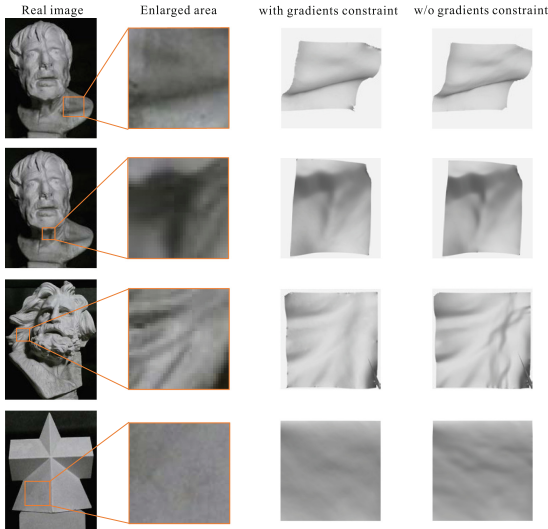
### 3 Experiments

#### 3.1 Setup

In this paper, we evaluate the proposed method with the uncalibrated PS method (LDR) [13] and a depth refinement method (SIG) based on single RGB image [12]. We collect twenty-five RGB images under different illuminations for 3 real scenes by a Kinect II depth camera, respectively. For the SIG method, we also average the depth map and form another baseline (SIG\_M). Since we have no ground truth depth map of the 3 scenes, we use two self-defined criteria to evaluate the refinement accuracy, one is a local evaluation of depth map including local variance (LE\_V) and local nuclear norm (LE\_N), the other one is comparing the outline similarity of refinement result with real image. Besides, we use the refined depth and camera intrinsic parameter to generate 3D reconstruction result and make a visual comparison of these methods.

#### 3.2 Quantitative Comparison

As shown in Fig. 3, our method can generate a high-resolution refined result for depth camera by transforming the depth map to RGB image. Besides, compared



**Fig. 4.** The comparison of the proposed method with and without multi-shading gradients constraint. The last two columns show the local 3D reconstruction results.

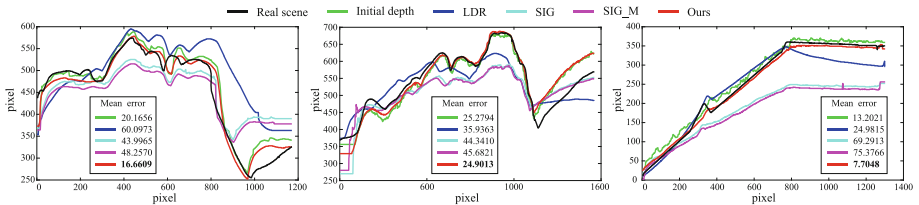
with the enlarged areas of RGB images captured by Kinect II, Fig. 4 shows the visual comparison of our method with and without the multi-shading gradients constraint. We can clearly see that, the multi-shading gradients can effectively restrict the 3D reconstruction results more similar to real scenes.

We use LE\_V and LE\_N to show the local reconstruction performance, all the two criterion indicate the local noise level of the depth. The results are shown in Table 1. We can clearly see that our method and LDR have lower scores than the other methods. In fact, refer to Fig. 1, photometric stereo method generates accurate local details but inaccurate global result, so it is reasonable that some evaluation results of LDR is higher than ours in Table 1.

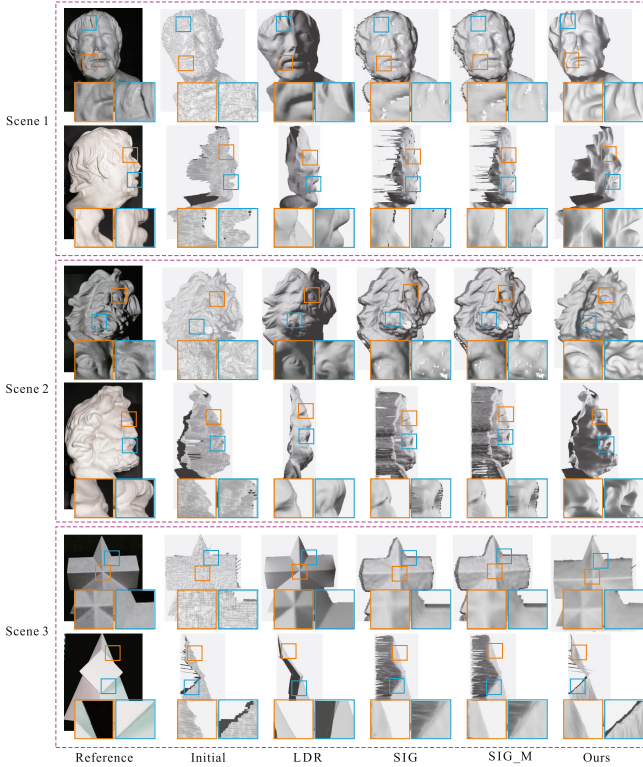
**Table 1.** The mean local variance (LE\_V) and mean local nuclear norm (LE\_N) of the refinement depths for different methods.

Method	Initial depth	LDR	SIG	SIG_M	Ours	
Scene 1	LE_V	39.3343	4.7929	38.4422	37.1902	4.0622
	LE_N	16870.38	882.94	16828.82	16844.71	12913.72
Scene 2	LE_V	20.2676	4.0108	19.4531	18.8557	4.7397
	LE_N	15042.71	837.19	15079.18	15072.27	12499.90
Scene 3	LE_V	3.3966	2.9783	2.9803	3.03266	2.6599
	LE_N	16687.28	688.80	16665.95	16672.49	13476.77

For evaluating the global depth refinement accuracy of all methods, we compare the outline similarity of 3D reconstruction results of these methods with the real image. We align the profiles of 3D surfaces to the real image coordinate, Fig. 5 shows the outline similarity comparison results of the 3 scenes, and the mean error of each method, i.e., the pixel displacement is also shown. We can find that the proposed method has better outline similarity than the other methods. Note, the initial depth map of depth camera can also generate a good outline similarity result, but can not reproduce the surface details well, e.g., there exist many local noises as shown in the third scene of Fig. 5.



**Fig. 5.** Outline similarity comparison of different methods.



**Fig. 6.** More depth refinement results of our approach and baseline methods.

Figure 6 shows the visual comparison of our method and baselines. Note, for the PS method LDR, we generate depth map from the normal results. The first column is one RGB image captured by Kinect II depth camera, the second column indicates the 3D reconstruction results based on initial depth map, the last column denotes the results of our method. We can clearly see that our method can generate higher accurate 3D reconstruction result.

## 4 Conclusion

In this paper, we have proposed a feasible high-resolution depth refinement approach, by effectively combining both photometric and multi-shading gradients constraints. The photometric constraint refines the local details of scene surface and the multi-shading gradients constraint guarantees to suppress the influence of inaccurate approximation from single illumination image. Extensive experimental results validate that the proposed approach can always generate more accurate depth map and thereby the 3D reconstruction results with higher resolution and richer local structural details. In the future, we are interested in

further exploring how to combine multi-view stereo in our model to acquire even better depth refinement performance. We also want to seek faster optimization solution to our energy function to enable near real-time response.

## References

1. Ahmed, A.H., Farag, A.A.: Shape from shading under various imaging conditions. In: CVPR (2007)
2. Bai, J., Yang, J., Ye, X., Hou, C.: Depth refinement for binocular kinect RGB-D cameras. In: VCIP (2016)
3. Chatterjee, A., Govindu, V.M.: Photometric refinement of depth maps for multi-albedo objects. In: CVPR (2015)
4. Feng, W., Tian, F., Zhang, Q., Sun, J.: 6D dynamic camera relocalization from single reference image. In: CVPR (2016)
5. Feng, W., Tian, F., Zhang, Q., Zhang, N., Wan, L., Sun, J.: Fine-grained change detection of misaligned scenes with varied illuminations. In: ICCV (2015)
6. Han, Y., Lee, J.Y., So Kweon, I.: High quality shape from a single RGB-D image under uncalibrated natural illumination. In: ICCV (2013)
7. Haque, M., Chatterjee, A., Madhav Govindu, V.: High quality photometric reconstruction using a depth camera. In: CVPR (2014)
8. Horn, B.K., Brooks, M.J.: The variational approach to shape from shading. *Comput. Vis. Graph. Image Process.* **33**(2), 174–208 (1986)
9. Liu, S., Do, M.N.: Inverse rendering and relighting from multiple color plus depth images. *IEEE TIP* **26**(10), 4951–4961 (2017)
10. Maier, R., Kim, K., Cremers, D., Kautz, J., Niener, M.: Intrinsic3D: high-quality 3D reconstruction by joint appearance and geometry optimization with spatially-varying lighting. In: ICCV (2017)
11. Miao, D., Tian, F., Feng, W.: Active camera relocalization with RGBD camera from a single 2D image. In: ICASSP (2018)
12. Or-El, R., Rosman, G., Wetzler, A., Kimmel, R., Bruckstein, A.M.: RGBD-fusion: real-time high precision depth recovery. In: CVPR (2015)
13. Papadhimitri, T., Favaro, P.: A closed-form, consistent and robust solution to uncalibrated photometric stereo via local diffuse reflectance maxima. *IJCV* **107**(2), 139–154 (2014)
14. Park, J., Sinha, S.N., Matsushita, Y., Tai, Y.W., Kweon, I.S.: Multiview photometric stereo using planar mesh parameterization. In: ICCV (2013)
15. Shi, Y., Tian, F., Miao, D., Feng, W.: Fast and reliable computational rephotography on mobile device. In: ICME (2018)
16. Woodham, R.J.: Photometric method for determining surface orientation from multiple images. *Opt. Eng.* **19**(1), 1–22 (1980)
17. Wu, C., Zollhfer, M., Niener, M., Stamminger, M., Izadi, S., Theobalt, C.: Real-time shading-based refinement for consumer depth cameras. *ACM TOG* **33**(6), 200 (2014)
18. Zhang, Q., Feng, W., Wan, L., Tian, F., Tan, P.: Active recurrence of lighting condition for fine-grained change detection. In: IJCAI (2018)
19. Zhang, Q., Ye, M., Yang, R., Matsushita, Y., Wilburn, B., Yu, H.: Edge-preserving photometric stereo via depth fusion. In: CVPR (2012)
20. Zhang, S., Wang, C., Chan, S.C.: A new high resolution depth map estimation system using stereo vision and kinect depth sensing. *J. Signal Process. Syst.* **79**(1), 19–31 (2015)



# Weakly-Supervised Object Localization by Cutting Background with Deep Reinforcement Learning

Wu Zheng<sup>1,2,4</sup> and Zhaoxiang Zhang<sup>1,2,3,4(✉)</sup>

<sup>1</sup> Research Center for Brain-inspired Intelligence, CASIA, Beijing, China

<sup>2</sup> National Laboratory of Pattern Recognition, CASIA, Beijing, China

<sup>3</sup> CAS Center for Excellence in Brain Science and Intelligence Technology,  
Beijing, China

<sup>4</sup> University of Chinese Academy of Sciences, Beijing, China  
{zhengwu2016,zhaoxiang.zhang}@ia.ac.cn

**Abstract.** Weakly-supervised object localization only depends on image-level labels to obtain object locations and attracts more attention recently. Taking inspiration from the human visual mechanism that human searches and localizes the region of interest by shrinking the view from a wide range and ignoring the unrelated background gradually, we propose a novel weakly-supervised localization method of cutting background of an object iteratively to achieve object localization with deep reinforcement learning. This approach can train an agent as a detector, which searches through the image and tries to cut off all regions unrelated to classification performance. An effective refinement approach is also proposed, which generates a heat-map by sum-pooling all feature maps to refine the location cropped by the agent. As a result, by combining the top-down cutting process and the bottom-up evidence for refinement, we can achieve a good performance on object localization in only several steps. To the best of our knowledge, this may be the first attempt to apply deep reinforcement learning to weakly-supervised object localization. We perform our experiments on PASCAL VOC dataset and the results show our method is effective.

**Keywords:** Weakly-supervised object localization  
Deep reinforcement learning · Convolutional neural network

## 1 Introduction

The current state-of-art localization results come from approaches of fully supervision, such as [1–5]. Fully supervision means providing both bounding boxes and labels of objects in the image during training. However, labelling the samples is time-consuming and expensive, which limits the usability of localization task significantly. In contrast, weakly-supervised object localization [7–10, 12, 21] does not require annotated bounding boxes but only the image-level labels.

Though such methods are usually less accurate than fully-supervised methods, it is often considered as an acceptable sacrifice to reduce dependency for annotated datasets.



**Fig. 1.** An simplified demonstration of our localization process.

Exciting recent weakly-supervised object localization methods [6, 13–15] has shown that the discriminative object part can be localized using class activation map (CAM) [6], which is a kind of heat-map generated by grouping class-specific convolutional feature maps. However, these methods usually generate large amounts of candidate proposals based on CAM and select the candidate with highest confidence as target location, which is time-consuming and inconsistent with human visual mechanism.

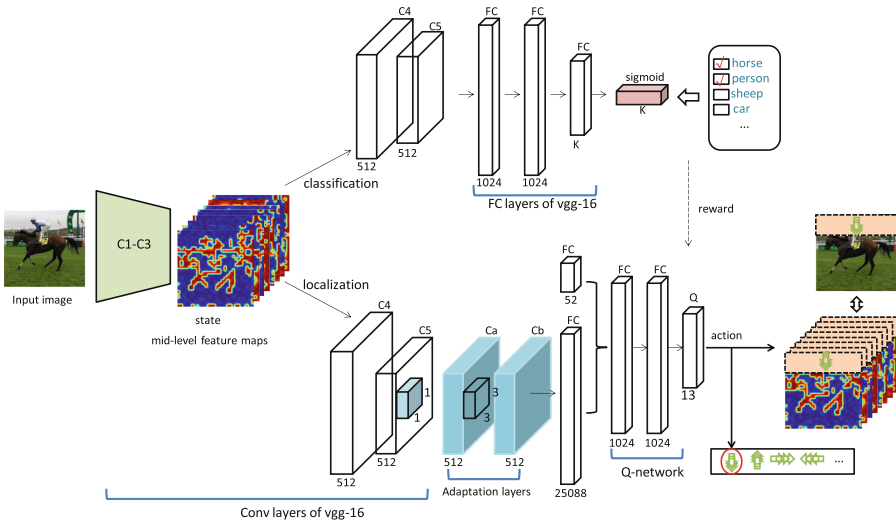
By contrast, our proposed method tries to seek a different approach to achieve weakly-supervised object localization. Taking inspiration from the human visual mechanism that human searches and localizes the region of interest by shrinking the view from a wide range and ignoring unrelated background gradually, we propose a novel weakly-supervised localization method of cutting background of an object iteratively to obtain object locations with deep reinforcement learning. We train an agent as a detector, which searches through the image and tries to cut off all regions unrelated to classification performance, as depicted in Fig. 1. To achieve better localization performances, we propose a bottom-up approach that sum-pooling all feature maps to generate a heat-map for refining the cropped result. Compared with previous CAM-based methods, our approach conforms more to the human visual mechanism and can localize the object intelligently. To the best of our knowledge, this may be the first attempt to apply deep reinforcement learning to weakly-supervised object localization. We believe that it may provide a brand new perspective to address this problem. Overall, our contributions can be summarized as follows:

- We propose a novel deep reinforcement learning approach to achieve weakly-supervised object localization. Compared with previous methods, we can crop the object location quickly in several steps without generating large amount of region candidates and selecting the best one.
- We propose a new approach to refine the predictions of weakly-supervised object localization and improve the localization performance.

## 2 Methodology

### 2.1 Overview

We aim to achieve weakly-supervised object localization by training an agent with deep reinforcement learning for cutting background intelligently. Firstly, we replace the last softmax layer of the vgg-16 pretrained on [20] with a sigmoid layer and fine-tune it for performing multi-label classification. Secondly, we train a deep Q-network (DQN) [16] for cutting unrelated background to achieve object localization. Finally, the location cropped by the agent will be refined by the heat-map generated by sum-pooling the feature maps.



**Fig. 2.** Overview of the proposed weakly-supervised object localization framework.

### 2.2 Deep Reinforcement Learning for Localization

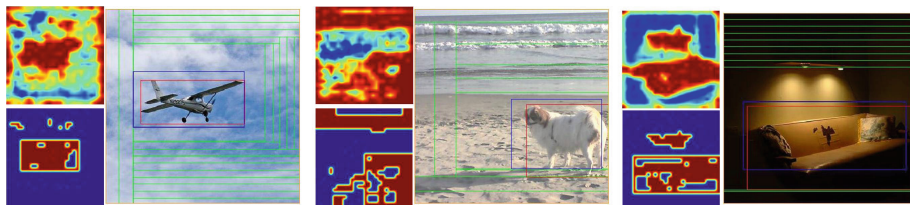
The proposed framework is depicted in Fig. 2. We perform the cutting actions on the mid-level feature maps due to a trade-off between the speed and fineness of cutting. In the training process, there are two streams in our framework, the upper stream is the modified vgg-16 for multi-label classification and the lower stream is a DQN for localization, which takes advantage of the last two group convolutional layers of vgg-16 for extracting senior features. To adapt to the task of DQN, we add two convolutional adaptation layers following the final convolutional layers of vgg-16. In the testing process, we just use the lower stream to achieve weakly-supervised object localization. Next we will detail on the reinforcement learning settings.



**State.** The state representation consists of two elements. One is a vector of senior feature information extracted from the cropped mid-level feature maps, we pad the cut-off region of mid-level feature maps with zeros to keep the aspect ratio unchanged and so avoid the dramatic decrease of classification performance. We cut feature maps on the dimensions of width and height and apply the cutting action to all channels. The other element is a vector of 4 past executed actions, which are encoded as an one-hot vector.



**Fig. 3.** Examples of object localized by the agent. The green, blue, and red lines show the cutting process, final cropped results and ground truth bounding boxes separately. (Color figure online)



**Fig. 4.** Examples of refinement with heat-maps. In each group pictures, the left-top and left-bottom pictures show the original heat-map and the heat-map with the drawn maximal region. In the right picture, the smallest green rectangle and the blue rectangle represent the prediction of agent and final result refined by the heat-map, respectively. (Color figure online)

**Action.** There are two types of actions: cutting actions for cutting feature maps and terminal action for terminating the cutting process. The cutting actions involve different directions and scales, which means we can crop the square feature map from the four sides  $\{up, down, left, right\}$  in three scales  $\{\frac{1}{28}, \frac{2}{28}, \frac{3}{28}\}$ . Combining multiple scales of actions helps us balance the speed and accuracy of cutting. There are two cases of terminating the cutting process, one is that the terminal action is selected by the agent, the other is that the classification

score of the cropped feature maps is less than the pre-defined threshold. Thus we have totally 13 actions for agent to select. Figure 3 gives some examples of object localization process based on the action space.

**Reward.** In weakly-supervised setting, we can roughly judge whether the object is retained or cut off only from the classification score. However, sometimes cutting off the specific background will result in a significant decrease of classification score, while cutting off parts of an object will have no effect. Therefore, we define a descent threshold of classification score to balance when to cut or stop. And we set different scales of rewards corresponding to different scales of actions, e.g. rewards = 1, 2, 3 for actions = 1, 2, 3, which aims to encourage large scales of cutting actions in the early stage to finish the cutting process as soon as possible, and encourage small scales of actions in the late stage to get a accurate result. Let denote the classification score of the original and the cropped feature maps as  $\bar{p}(c|x)$  and  $\hat{p}(c|x)$ , respectively. Denote the threshold as  $\delta$ . Then the reward function for non-termination case is

$$R(s, s') = \begin{cases} +scale, & \bar{p} - \hat{p} \leq \delta, \\ -scale, & \text{otherwise.} \end{cases} \quad (1)$$

For termination case, the reward function is

$$R(s) = \begin{cases} +\eta, & \bar{p} - \hat{p} \leq \delta, \\ -\eta, & \text{otherwise.} \end{cases} \quad (2)$$

The scalar  $\eta$  represents the absolute value size of reward for termination action, it is a hyperparameter.

**Q-learning.** Q-learning is a reinforcement learning algorithm, which can train the agent to select the optimal action according to a specific state. We build a DQN to estimate the action-state value function  $Q(s, a)$ , which is an approximation of the expected rewards after executing the action  $a$  on current state  $s$ . The agent will choose the action corresponding to the maximal  $Q$  value.  $\gamma$  is a discount factor, then the  $Q(s, a)$  and its update rule are denoted as

$$Q(s, a) = r + \gamma \max_{a'} Q(s', a') \quad (3)$$

$$Q(s, a) = Q(s, a) + \alpha[r + \gamma \max_{a'} Q(s', a') - Q(s, a)] \quad (4)$$

### 2.3 Refinement

Motivated by [17], we think that the heat-map generated by sum-pooling all feature maps may provide extra information beyond classification score, which can be used to refine the result cropped by the agent. As depicted in Fig. 4, we find the maximal response region of heat-map can cover the object itself very well in most cases. Thus, we firstly convert the heat-map to a binary image according and draw a rectangle to surround the area of maximal response, the thresholds used to generate binary image are evaluated on a small validation subset. Then we take the intersection between the cropped location and the rectangle of the maximal response region in heat-map as our final result.

### 3 Experiment

In our experiments, we firstly fine-tune the modified vgg-16 for multi-label classification on the VOC2007 trainval and VOC2012 training set, and test it on the VOC2007 [18] test set and VOC2012 [19] validation set, with results summarized in Table 1. Secondly, we train the DQN based on the trained vgg-16 and test it for localization on VOC2007 test set and VOC2012 validation set, with results summarized in Tables 2 and 3. Finally, we explore the effects of our proposed refinement method, and the results are summarized in Table 4.

#### 3.1 Network Training

We replace the last softmax layer of pre-trained vgg-16 with sigmoid layer and fine-tune it on VOC2007 trainval set+VOC2012 training set for multi-label classification. We train each agent for 50 epoches. In the training process, we use RMSProp optimizer with a decaying learning rate ranging from  $1e-5$  to  $1e-6$  in the early 10 epoches. We use the  $\epsilon$ -greedy policy to explore more state-action pairs and the  $\epsilon$  is decayed in steps of 0.1 from 1 to 0.1 over the first 10 epoches. Discount factor  $\gamma$  is set to 0.95. The descent threshold  $\delta$  of classification score for each class is evaluated on a small validation set.

**Table 1.** Average precisions of multi-label classification tested on VOC2007 test set and VOC2012 validation set.

Category	plane	bike	bird	boat	btl	bus	car	cat	chair	cow	table	dog	horse	moto	pers	plant	sheep	sofa	train	tv	mAP
Accuracy-voc07	98.7	85.6	94.3	89.7	66.3	89.4	83.7	96.4	73.7	84.4	79.3	95.1	94.3	92.6	95.9	58.7	87.5	69.8	95.7	82.3	85.7
Accuracy-voc12	98.6	85.5	94.4	89.6	66.4	89.5	83.5	96.3	73.6	84.2	79.4	94.9	94.5	92.5	95.8	58.5	87.3	70.0	95.5	82.2	85.6

**Table 2.** Horizontal comparison of average precisions for object localization on VOC2007 test set.

Method	mAP
Wang [10]	<b>30.9</b>
Bency [17]	25.7
Gudi [15]	30.2
Ours	<b>30.5</b>

**Table 3.** Horizontal comparison of average precisions for object localization on VOC2012 validation set.

Method	mAP
Qquab [13]	11.7
Bency [17]	26.5
Gudi [15]	25.4
Ours	<b>28.8</b>

### 3.2 Localization Prediction Metric

We use the standard object detection bounding box overlap metric Intersection-Over-Union (IOU) to determine correctness of the predicted location. If the IOU between the predicted location and the ground truth bounding box exceeds 0.5, the predicted location will be labeled as correct. Otherwise, we count the prediction as a false positive and increment the false negative count. We define the confidence of the predicted location as its classification score. The average precision is calculated according to the standard algorithm.

### 3.3 Performance and Analysis

**Classification Performance.** Table 1 concludes the results of multi-label classification on VOC2007 test set and VOC2012 validation set. We can see that the average precisions of most categories exceed 80%, which makes a good basis for our localization experiment.

**Localization Performance.** Localization results are summarized in Tables 2 and 3. The average precision of our method is computed by localizing one object of same category per image while baselines are for multiple objects detection, thus we make a horizontal comparison. We find the proposed approach achieves a competitive results with baselines, which indicates the validity of our method.

**Table 4.** mAP of refinement on VOC2007 test set and VOC2012 validation set.

Dataset	No refinement	Refinement
VOC2007	25.2	<b>30.5</b>
VOC2012	24.4	<b>28.8</b>

**Localization Refinement.** We show the refinement results on VOC2007 test set and VOC2012 validation set in Table 4. Our refinement method is based on the location information in heat-map. We take the intersection between this maximal region in heat-map and the location cropped by the agent as our final result, which aims to remove some background outside the rectangle of maximal response region. It bring a 4-5% improvement to the performance of agent.

**Table 5.** Comparison of number of proposals.

Model	Proposals
SS-Based	2000
Ours	9

**Proposals Analysis.** Many recent methods are based on CAM, like [15], and use the unsupervised method like Selective Search [11] to generate large amount of candidate proposals, with each proposal to be classified separately to determine as the positive or negative sample. As shown in Table 5, compared to the methods that generate about 2000 proposals for each image, our proposed method generate average 9 proposals and do 9 times classification per image. Therefore, our approach are much faster than those based on the unsupervised method, which can save large amount of time in detection process. Besides, our search process guided by reinforcement learning is more human-like than the unsupervised exhaustive search methods.

## 4 Conclusion

This paper presents a deep reinforcement learning solution to weakly-supervised object localization by cutting background iteratively, which is more human-like and consistent with human visual mechanism compared with those methods depending on unsupervised region proposals. Also, by combining the top-down cutting process and bottom-up refinement, we can cut object background out intelligently only in several steps and achieve a good localization performance, which is much faster than those methods based on unsupervised generation of proposals. We believe it may provide a brand new perspective to address weakly-supervised object localization.

**Acknowledgement.** This work was supported in part by the National Key R&D Program of China(No. 2018YFB1004600), the National Natural Science Foundation of China (No. 61773375, No. 61375036, No. 61602481, No. 61702510), and in part by the Microsoft Collaborative Research Project.

## References

1. Girshick, R.: Fast R-CNN. In: Computer Science (2015)
2. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: CVPR (2014)
3. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., Berg, A.C.: SSD: Single shot MultiBox detector. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 21–37. Springer, Cham (2016). [https://doi.org/10.1007/978-3-319-46448-0\\_2](https://doi.org/10.1007/978-3-319-46448-0_2)
4. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: unified, real-time object detection. In: CVPR (2016)
5. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **39**(6), 1137–1149 (2015)
6. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: CVPR (2016)
7. Li, D., Huang, J.B., Li, Y., Wang, S., Yang, M.H.: Weakly supervised object localization with progressive domain adaptation. In: CVPR (2016)

8. Cinbis, R.G., Verbeek, J., Schmid, C.: Weakly supervised object localization with multi-fold multiple instance learning. *IEEE Trans. Pattern Anal. Mach. Intell.* **39**(1), 189 (2015)
9. Song, H.O., Girshick, R., Jegelka, S., Mairal, J., Harchaoui, Z., Darrell, T.: On learning to localize objects with minimal supervision. arXiv preprint [arXiv:1403.1024](https://arxiv.org/abs/1403.1024) (2014)
10. Wang, C., Ren, W., Huang, K., Tan, T.: Weakly supervised object localization with latent category learning. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) *ECCV 2014*. LNCS, vol. 8694, pp. 431–445. Springer, Cham (2014). [https://doi.org/10.1007/978-3-319-10599-4\\_28](https://doi.org/10.1007/978-3-319-10599-4_28)
11. Uijlings, J.R., Sande, K.E., Gevers, T., Smeulders, A.W.: Selective search for object recognition. *Int. J. Comput. Vis.* **104**(2), 154–171 (2013)
12. Bilen, H., Vedaldi, A.: Weakly supervised deep detection networks. In: *CVPR* (2016)
13. Oquab, M., Bottou, L., Laptev, I., Sivic, J.: Is object localization for free?-weakly-supervised learning with convolutional neural networks. In: *CVPR* (2015)
14. Durand, T., Mordan, T., Thome, N., Cord, M.: WILDCAT: weakly supervised learning of deep ConvNets for image classification, pointwise localization and segmentation. In: *CVPR* (2017)
15. Gudi, A., van Rosmalen, N., Loog, M., van Gemert, J.: Object-extent pooling for weakly supervised single-shot localization. arXiv preprint [arXiv:1707.06180](https://arxiv.org/abs/1707.06180) (2017)
16. Mnih, V., et al.: Playing Atari with deep reinforcement learning. arXiv preprint [arXiv:1312.5602](https://arxiv.org/abs/1312.5602) (2013)
17. Bency, A.J., Kwon, H., Lee, H., Karthikeyan, S., Manjunath, B.S.: Weakly supervised localization using deep feature maps. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) *ECCV 2016*. LNCS, vol. 9905, pp. 714–731. Springer, Cham (2016). [https://doi.org/10.1007/978-3-319-46448-0\\_43](https://doi.org/10.1007/978-3-319-46448-0_43)
18. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL visual object classes challenge 2007 (VOC 2007) results. <http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html>
19. Everingham, M., Van Gool, L., Williams, C., Winn, J., Zisserman, A.: The pascal visual object classes challenge 2012 (voc 2012) results (2012). <http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html>
20. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Li, F.F.: Imagenet: a large-scale hierarchical image database. In: *CVPR*, pp. 248–255 (2009)
21. Gokberk Cinbis, R., Verbeek, J., Schmid, C.: Multi-fold mil training for weakly supervised object localization. In: *CVPR* (2014)



# Nature-Inspired Computational Model for Solving Bi-objective Traveling Salesman Problems

Xuejiao Chen, Zhengpeng Chen, Yingchu Xin, Xianghua Li, and Chao Gao<sup>(✉)</sup>

School of Computer and Information Science,  
Southwest University, Chongqing 400715, China  
cgao@swu.edu.cn

**Abstract.** Bi-objective Traveling Salesman Problem (BTSP) is an NP-hard problem in the combinatorial optimization, which is also important question in the field of operations research and theoretical computer science. Genetic Algorithm (GA) is one type of efficient methods for solving NP-hard problems. However, GA-based algorithms suffer high time computational complexity, low stability and the premature convergence for solving BTSP. This paper proposes an improved method of genetic algorithm based on a novel nature-inspired computational model to solve these problems. The initialization of population of proposed algorithm is first optimized by the prior knowledge of *Physarum*-inspired computational model (PCM) in order to enhance the computational speed and stability. Then the hill climbing method (HC) is used to increase the diversity of the individuals and avoid falling into the local optimum. A series of experiments are conducted and results show that our proposed algorithm can achieve the better performance.

**Keywords:** Bi-objective traveling salesman problem · NSGA-II  
Hill climbing · *Physarum*

## 1 Introduction

Multi-objective problems (MOPs) are widespread in the real world, such as the multi-objective network structure design and multi-objective scheduling problems in the flow shop. It's a challenging problem for us to design an efficient algorithm to provide effective solutions. Lots of real-world problems can be formulated as a multi-objective traveling salesman problem (MOTSP) [1].

In the past few decades, evolutionary algorithms (EAs) have been widely applied for solving multi-objective traveling salesman problems, which can combine the ability of global exploitation and local refinement. One of the most well-known methods is NSGA-II [2], which incorporates an elitism preservation strategy in evolutionary algorithms. Some studies in different test problems have shown that NSGA-II is able to maintain a better spread of solutions and converge better in the obtained nondominated front. NSGA-II for solving BTSP,

however, often cannot achieve a good trade-off solution or suffers premature convergence, due to the disturbance of non-global optimal paths. Taking NSGA-II as a benchmark algorithm, a new nature-inspired computational model is applied to optimize the distribution of initialization population in order to help the original algorithm overcome the above problems.

Currently, a series of biological experiments have demonstrated that a unicellular and multi-headed slime mold, *Physarum* shows an ability to solve mazes and construct efficient and robust networks [4]. Tero et al. have formulated the positive feedback mechanism of *Physarum* in foraging [3]. Gao et al. further have described the characteristics of *Physarum* from the bionic mechanism model and intelligent computation [4]. Based on the positive feedback mechanism, PCM can generate the raw material pipes to link the maze with the shortest path. Based on such prior knowledge, a *Physarum*-inspired model is proposed to optimize the initialization of population of NSGA-II for solving multi-objective traveling salesman problems. Meanwhile, the hill climbing method (HC) is used to increase the diversity of the individuals and avoid falling into the local optimum.

## 2 Related Work

### 2.1 Basic Concepts of MOOP and Pareto-Optimal Solutions

A multi-objective optimization problem (MOOP) aims to deal with two or more objective functions simultaneously. As usual, a MOOP which satisfies the  $p$  inequality constraints and  $q$  equality constraints can be formulated as Eq. (1).

$$MOOP = \begin{cases} \min F(X) = (f_1(x), \dots, f_m(x)); & \text{for } m = 2, \dots, M \\ \text{subject to } G_i(X) > 0; & \text{for } i = 1, 2, \dots, p \\ H_j(X) = 0; & \text{for } j = 1, 2, \dots, q \end{cases} \quad (1)$$

Since different objectives in MOOP are usually conflicting, it is very difficult to compare with solutions obtained by different objectives. It is quite hard to find the best solution that can optimize all objectives simultaneously. Instead, there still are a number of solutions in the solution space in which no solution is superior to others for all objectives. The goal of MOOP is to obtain these non-dominated solutions with good trade-offs among different objectives which are named as Pareto set. The related definitions [5] are as follows.

**Definition 1.** A solution  $X_1 = (x_1^1, x_1^2, \dots, x_1^n)^T$  is said to be dominated by  $X_2 = (x_2^1, x_2^2, \dots, x_2^n)^T$ , denoted as  $X_2 \prec X_1$ , if both conditions mentioned below are satisfied:

$$\begin{aligned} \forall i \in (1, 2, \dots, k) : f_i(X_1) &\leq f_i(X_2) \\ \exists i \in (1, 2, \dots, k) : f_i(X_1) &< f_i(X_2) \end{aligned} \quad (2)$$



**Definition 2.** If a solution is not dominated by any other solutions in feasible solution set, then it is named as a Pareto optimal solution or non-dominated solution. The set of all Pareto optimal solutions is named as a Pareto set (PS), i.e.

$$PS = \{X \in D | \nexists Y \in D, F(Y) \prec F(X)\} \tag{3}$$

**Definition 3.** The objective vector corresponding to PS in the objective space is named as the Pareto front (PF).

$$PF = \{F(X) | X \in PS\} \tag{4}$$

For MOOP instances, a true PS is always unknown [6]. Instead, the pseudo-optimal PS is defined as an approximation of the true PS, which is obtained by fusing all PSs returned by all existing algorithms in several runnings [7].

### 2.2 The Definition of BTSP

As an extension of a single objective TSP, BTSP manages two objectives simultaneously. The BTSP consists in finding a Hamiltonian cycle of  $N$  cities that optimizes the following minimization problem:

$$h_k(x) = C_{(m(N),m(1))}^k + \sum_{i=1}^{N-1} C_{(m(i),m(i+1))}^k, k = 1, 2 \tag{5}$$

where  $m(i)$  refers to the  $i^{th}$  city,  $C_{(m(i),m(j))}^k$  represents the value factor between  $m(i)$  and  $m(j)$  for an objective  $k$ .

Four typical measurements based on the definition of Garca-Martinez [8] are used to estimate the performances of BTSP algorithms:

- (1) The graphical representation of PF is returned by an algorithm. These graphics provide a visual information for estimating the quality and distribution of solutions. It is an intuitive measurement of PF with a graphical representation. If there are two PFs, PFA and PFB, and the results of PFA converge to the bottom-left region comparing with those of PFB, we can deduce that the results of PFA are better than those of PFB.
- (2)  $M_1$  metric represents the distance between the results of an algorithm, denoted as  $Y$ , and the pseudo-optimal Pareto front ( $\bar{Y}$ ). This metric is based on Eq. (6), in which  $|Y|$  means the number of non-dominated solutions in front of  $Y$ . The smaller  $M_1$  metric is, the smaller difference between  $\bar{Y}$  and  $Y$  is.

$$M_1(Y) = \frac{1}{|Y|} \sum_{p \in Y} \min\{\| P - \bar{P} \|; \bar{P} \in \bar{Y}\} \tag{6}$$

- (3)  $M_2$  metric evaluates the distribution of solutions in PF returned by an algorithm (denoted as  $Y$ ). This metric is based on Eq. (7), in which the parameter is a positive constant. The larger  $M_2$  metric is, the wider the coverage of the obtained solutions is.

$$M_2(Y) = \frac{1}{|Y - 1|} \sum_{p \in Y} |\{q \in Y; \|p - q\| > \sigma\}| \quad (7)$$

- (4)  $M_3$  metric is used to evaluate the diameter of PF returned by an algorithm (denoted as  $Y$ ) based on Eq. (8), in which  $p_i$  denotes the solution value in  $p$  for objective  $i$ . The larger  $M_3$  metric is, the larger region of the objective space of solutions locates.

$$M_3(Y) = \sqrt{\sum_{i=1}^2 \max\{\|p_i - q_i\|; p, q \in Y\}} \quad (8)$$

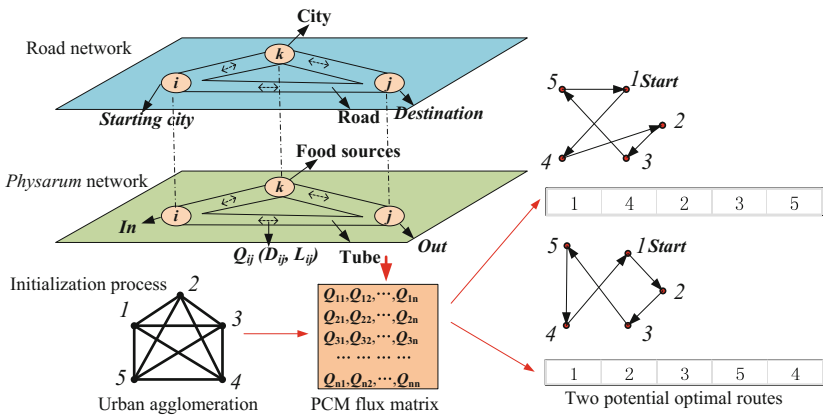
### 3 *Physarum*-inspired NSGA\_II for BTSP

#### 3.1 The Formulation of GA-based BTSP Method

GA-based methods are flexible methods that can be used, in principle, to solve various types of problems. During the optimization process, there are some steps in GA-based methods, i.e., population initialization, crossover and mutation operators, for improving the value of a criterion and escaping from the local minima. Existing studies on GA-based algorithms focus on optimizing encoding, selection, crossover and mutation operations. The approach which involves both improved population initialization and local search operators is rarely considered in the literature. In this paper, we take advantage of the nature-inspired computational model and the hill climbing (HC) to improve the population initialization and escape from the local minima. More specifically, taking NSGA\_II as a benchmark algorithm, a new *Physarum*-based network computational model (PCM) is applied to optimize the distribution of initialization population and increase the diversity of the population. NSGA\_II is an effective algorithm which incorporates an elitism preservation strategy in evolutionary algorithms [2]. The main idea of NSGA\_II is to reproduce a population by a genetic operator and then sort them based on the non-domination rank and crowding distance. It should be pointed that this algorithm tends to the premature convergence, high time computational complexity and low stability for solving BTSP. Therefore, this paper takes advantage of the proposed optimization strategy to improve the performance of NSGA\_II.

### 3.2 The Formulation of *p*NSGA-II

Taking advantage of PCM in solving path-finding problems, we propose two optimization methods to improve the efficiency of NSGA-II when solving a BTSP. The proposed algorithm is denoted as *p*NSGA-II. In the *p*NSGA-II, a *Physarum* network is mapped to the topology of a network. The food sources and tubes of *Physarum* network are defined as cities and paths connecting two different cities, respectively. We exploit the prior knowledge of *Physarum* network conductivity matrix to initialize the population. The optimized strategy can improve the search abilities of algorithms, from the following two perspectives. First, if the *Physarum* network conductivity matrices are consistent with the optimal solutions, optimized algorithm can reduce the searching space so as to speed up the convergence. Second, the *Physarum* network conductivity matrices can promote the population diversity in the same way.



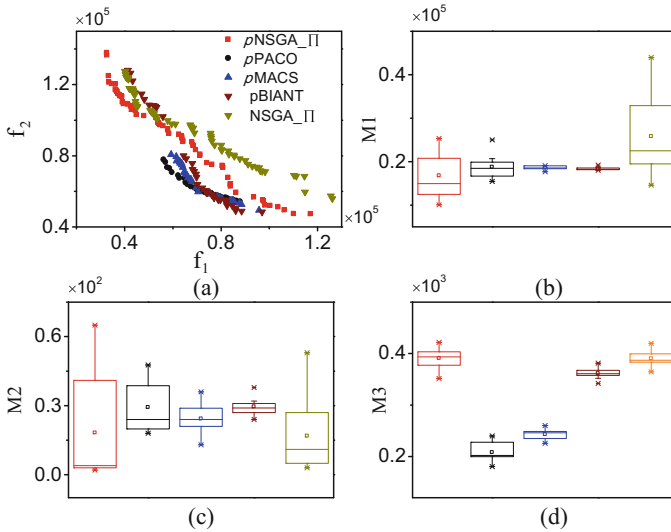
**Fig. 1.** The illustration of initialization process of *p*NSGA-II. The figure shows the food sources and tubes of *Physarum* network which represent cities and paths or cost in a road network, respectively. Taking the city agglomeration of size 5, each objective matrix generates the corresponding flow matrix using the positive feedback mechanism of PCM, and then the initial solution is constructed according to the flow matrix of corresponding objective.

Compared with the original NSGA-II, the main framework of the algorithms remain the same. The typical differences between the optimized and the original algorithm are the processes of initialization population and genetic operation. During the initialization, the population is preset with the priori knowledge of PCM. Figure 1 gives an example of the initialization process of city agglomeration with 5 nodes. In order to increase the diversity of individuals, the hill climbing method (HC) is added to the genetic operation of *p*NSGA-II. Given a chromosome  $C_k = \{C_k^1, C_k^2, \dots, C_k^n\}$ , a node  $C_k^i$  is randomly selected from  $C_k$  and then we replace the location  $i$  with a random location  $j$ , where  $j \neq i$ . Compared with the original chromosome, the new generated one is retained if it can achieve a better solution.

## 4 Experiments

### 4.1 Datasets and Parameters

The bi-objective symmetric TSP instances are obtained from the web page<sup>1</sup>. Each of these instances is constructed from two different single objective TSP instances with the same number of nodes. In this paper, for the sake of contrastive analysis, we use the medium scale and larger scale bi-objective TSP instances, i.e., euclidAB100, kroAB150, to estimate our proposed method.



**Fig. 2.** Comparison results of different algorithms in EuculidAB100 instances in terms of (a) PF, (b) M1, (c) M2 and (d) M3 index.  $f_1$  and  $f_2$  denote values in EuculidA100 and EuculidB100, respectively. Results show that the pNSGA\_II algorithm is better than other algorithms. Especially on the M1 and M3 index, the solutions produced by pNSGA\_II locate a larger region of the solutions objective space and the coverage of the solutions is wider.

The parameters are set as follows. The initial value of the conductivity of each tube is 1. The total runs affected by PCM are 30. The total steps of iteration are 500 or 1000, according to the scale of instance. The size of population, the rate of mutation, crossover, hill-climbing are set to 500, 0.2, 0.6, 0.4, respectively, which are the same with the setting in NSGA\_II [2, 9]. All experiments are implemented on PC with 3.2 GHz CPU, 4 GB RAM and Windows 7 OS.

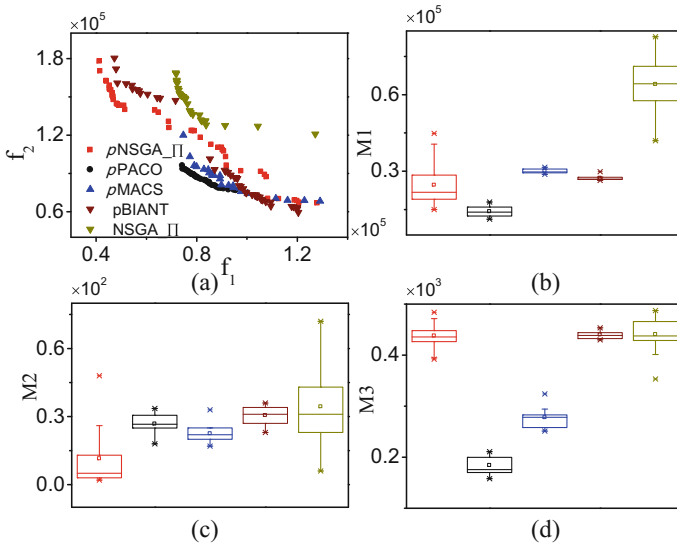
<sup>1</sup> <https://eden.dei.uc.pt/~paquete/tsp/>.

### 4.2 Experimental Results

All experiments are executed in the same environment to enable fair comparisons between our algorithm and other algorithms including NSGA\_II [2], *p*PACO, *p*MACS and *p*BIANT [9], which are the enhanced algorithms of PACO [10], MACS [11], BIANT [12], respectively. In order to wipe off the computational fluctuation, all results in our experiments are averaged over 30 times.

In schematic diagrams of PF, all PFs generated by each algorithm are aggregated into a single PF by removing the dominated solutions. In each box of M1, M2 and M3 index diagrams, the highest and lowest lines represent the maximum value and minimum value with 30 runnings, respectively. The upper and lower ends of a box are the upper and lower quartiles, respectively. The line within a box means the median of solutions.

Figure 2 plots the graphical representation of PFs, M1, M2 and M3 index returned by *p*NSGA\_II, *p*PACO, *p*MACS, *p*BIANT and NSGA\_II in EculidAB100 instances, respectively. This result shows that the optimized strategy for NSGA\_II can improve the quality and distribution of solutions, especially in M1 and M3 index. Such result represents that solutions produced by *p*NSGA\_II locate the larger region of the objective space and the coverage of obtained solutions is wider. According to Fig. 3, *p*NSGA\_II also shows better performance than others in KroAB150 instances.



**Fig. 3.** Comparison results in kroAB150 instances in terms of (a) PF, (b) M1, (c) M2 and (d) M3 index.  $f_1$  and  $f_2$  denote values in KroA150 and KroB150, respectively. Results show that *p*NSGA\_II is better than others. Especially on the M3 index, the solutions produced by *p*NSGA\_II locate a larger region of the objective space.

## 5 Conclusions

In this paper, we take advantage of the prior knowledge of PCM and the hill climbing method to optimize the initialization and genetic operators of GA-based optimization algorithm. Taking NSGA\_II as an example, the improved algorithm named as *p*NSGA\_II is applied to solve BTSP. Some experiments in bi-objective symmetric TSP instances are conducted in order to evaluate the efficiency of our proposed method. According to experimental results, we can conclude that the quality of solutions generated by *p*NSGA\_II is better than that of NSGA\_II, *p*PACO, *p*MACS and *p*BIANT.

**Acknowledgement.** This work is supported by the Fundamental Research Funds for the Central Universities (XDJK2016A008), the National Natural Science Foundation of China (61402379, 61403315).

## References

1. Fereidouni, S.: Solving traveling salesman problem by using a fuzzy multi-objective linear programming. *Afr. J. Math. Comput. Sci. Res.* **4**(11), 339–349 (2011)
2. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA\_II. *IEEE Trans. Evol. Comput.* **6**(2), 182–197 (2002)
3. Tero, A., et al.: Rules for biologically inspired adaptive network design. *Science* **327**(5964), 439–442 (2010)
4. Gao, C., et al.: Does being multi-headed make you better at solving problems? A survey of physarum-based models and computations. *Phy. Life. Rev.* <https://doi.org/10.1016/j.plev.2018.05.002> (2018)
5. Zhou, A.M., Qu, B.Y., Li, H., Zhao, S.Z., Suganthan, P.N., Zhang, Q.F.: Multi-objective evolutionary algorithms: a survey of the state of the art. *Swarm Evol. Comput.* **1**(1), 32–49 (2011)
6. Cao, Y.T., Smucker, B.J., Robinson, T.J.: On using the hypervolume indicator to compare Pareto fronts: applications to multi-criteria optimal experimental design. *J. Stat. Plan. Infer.* **160**, 60–74 (2015)
7. Chica, M., Cordon, O., Damas, S., Bautista, J.: Multiobjective constructive heuristics for the 1/3 variant of the time and space assembly line balancing problem: ACO and random greedy search. *Inform. Sci.* **180**(18), 3465–3487 (2010)
8. Garca-Martinez, C., Cordon, O., Herrera, F.: A taxonomy and an empirical analysis of multiple objective ant colony optimization algorithms for the bi-criteria TSP. *Eur. J. Oper. Res.* **180**, 116–148 (2007)
9. Zhang, Z.L., Gao, C., Lu, Y.X., Liu, Y.X., Liang, M.X.: Multi-objective ant colony optimization based on the *physarum*-inspired mathematical model for bi-objective traveling salesman problems. *PLoS ONE.* **11**(1), e0146709 (2016)
10. Doerner, K., Gutjahr, W.J., Hartl, R.F., Strauss, C., Stummer, C.: Pareto ant colony optimization: a metaheuristic approach to multiobjective portfolio selection. *Ann. Oper. Res.* **131**(1–4), 79–99 (2004)

11. Barn, B., Schaerer, M.: A multiobjective ant colony system for vehicle routing problem with time windows. In: Proceedings of Iasted International Multi-Conference on Applied Informatics, pp. 97–102 (2003)
12. Iredi, S., Merkle, D., Middendorf, M.: Bi-criterion optimization with multi colony ant algorithms. In: Zitzler, E., Thiele, L., Deb, K., Coello Coello, C.A., Corne, D. (eds.) EMO 2001. LNCS, vol. 1993, pp. 359–372. Springer, Heidelberg (2001). [https://doi.org/10.1007/3-540-44719-9\\_25](https://doi.org/10.1007/3-540-44719-9_25)



# Differential Evolution-Based Weighted Majority Voting for Crowdsourcing

Hao Zhang<sup>1</sup>, Liangxiao Jiang<sup>1,2</sup>(✉), and Wenqiang Xu<sup>1</sup>

<sup>1</sup> Department of Computer Science, China University of Geosciences,  
Wuhan 430074, China  
ljjiang@cug.edu.cn

<sup>2</sup> Hubei Key Laboratory of Intelligent Geo-Information Processing,  
China University of Geosciences, Wuhan 430074, China

**Abstract.** With the rapid development of crowdsourcing learning, inferring (integrating) truth labels from multiple noisy label sets, it is also called label integration, has been a hot research topic. And many methods have been proposed for label integration. However, due to the variable uncertainty of crowdsourced labelers, inferring truth labels from multiple noisy label sets still faces great challenges. In this paper we transform the label integration problem into an optimization problem, and exploit a differential evolution-based weighted majority voting method, simply DEWMV, for label integration. DEWMV searches and weights the voting quality of each label through the designed differential evolution (DE) algorithm. In DEWMV, we define three fitness functions, including the uncertainty of the integration label, the uncertainty of the class member probability and the hybrid uncertainty, to search the optimal voting quality for each label. By theoretically analyzing their effectiveness, we choose the hybrid uncertainty as the final fitness function for DEWMV. The experimental results on 14 real-world datasets show that DEWMV is superior to standard majority voting (MV) and all the other state-of-the-art label integration methods used to compare.

**Keywords:** Crowdsourcing · Multiple noisy labels · Label integration  
Label quality · Differential evolution

## 1 Introduction

Crowdsourcing [3] is a new approach to acquire class labels of instances from ordinary users on the Internet, which is more efficient and costs less than traditional way getting labels by professionals [5,6]. A crowdsourcing process mainly involves several sections: generating multiple noisy label sets by online labelers, inferring truth labels from multiple noisy label sets, noise filtering and correction, and learning from crowdsourced data. In view of the advantage of crowdsourcing, it has attracted much attention in the field of machine learning

---

This work was partially supported by NSFC (U1711267).



and data mining that need lots of labeled data. However, labels collected from crowdsourcing systems have various qualities and per instance in labeled data includes multiple noisy labels. The reason is that common labelers are greatly different in knowledge levels and evaluation standard of tasks. What's more, human error, bias, and so on are likely to have an impact. So how do we get the best integrated label from crowdsourced collected data?

Considerable research efforts have been devoted to improve the quality of data and use those noisy data. A common consensus approach is majority voting (MV) [5], which is simple but effective. In MV, the label which obtains the maximum number of votes is treated as the final integrated label. In addition to MV, many inference algorithms have been proposed to improve the quality of labels and the accuracy of integrated labels. These algorithms can be broadly classified into two categories according to their special assumptions. Some inference algorithms only follow two basic common assumptions: labelers have different reliabilities and independently make decisions, such as Dawid and Skene (DS) [1] and ZenCrowd (ZC) [2]. DS uses the maximum likelihood estimation to estimate a confusion matrices for each labeler and a class prior. ZC is a natural extension to MV, which uses only a two-element parameter to weight the reliability of a labeler. Other inference algorithms are designed based on some other additional assumptions, such as Raykar et al. (RY) [4] and a positive label threshold algorithm (PLAT) [9]. RY is a Bayesian approach to hypothesize that labelers have biases towards negative and positive instances which are named sensitivity and specificity. PLAT is proposed for dealing with the imbalanced labeling issue, which assumes labelers have different correction rates on negative and positive instances. Unfortunately, estimating the true labels for instances is still challenging. Especially, the labelers are adversarial or unreliable expertise and generally aim at the financial reward of annotation.

The purpose of this paper is to propose an efficient and effective method to weight the voting quality of labels in the process of inferring the true label for every instance. Although some existing label integration algorithms can estimate the label quality of crowdsourcing labelers, they are still a little crude and rarely use global intelligent optimization techniques to estimate the label quality of crowdsourcing labelers. In this paper we transform the label integration problem into an optimization problem, and exploit a differential evolution-based weighted majority voting method, simply DEWMV, for label integration. DEWMV searches and weights the voting quality of each label through the designed differential evolution (DE) algorithm. To search the optimal voting quality for each label, we define three fitness functions: (1) the uncertainty of the integration label (simply UIL); (2) the uncertainty of the class member probability (simply UCMP); (3) the hybrid uncertainty (simply HU). And then we choose the hybrid uncertainty as the final fitness function of DEWMV. The experimental results on 14 real-world datasets show that DEWMV significantly outperforms standard majority voting (MV) and all the other state-of-the-art label integration methods used to compare. At the same time our results illustrates that if the parameters are extremely suitable, simple parameters of labelers obviously upgrade the capability of inference algorithms.

The rest of the paper is organized as follows. Section 2 proposes our inference algorithm in detail. Section 3 describes experimental datasets and results. Section 4 concludes the paper.

## 2 Differential Evolution-Based Weighted Majority Voting

For a crowdsourcing system, the instance set is defined as  $\mathcal{E} = \{e_i\}_{i=1}^N$ , where each instance is  $e_i = \langle x_i, y_i, L_i \rangle$ ,  $x_i$  is the attribute vector,  $y_i$  is an unknown true label and  $L_i$  is a multiple noisy label set. In this paper, we don't consider the attributes of instance which is a binary classification problem. So each instance is simply defined as  $e_i = \langle y_i, L_i \rangle$  and  $y_i \in \{+, -\}$ . For instance  $e_i$ , multiple noisy label set  $L_i$  can be represented by a label vector  $\langle l_{i1}, \dots, l_{ij}, \dots, l_{ik} \rangle$ , where  $l_{ij}$  is the  $j$ th multiple noisy label and  $k$  is the number of multiple noisy labels of instance  $e_i$ . In the MV algorithm, the probability distribution of instance  $e_i$  that label type is binary belonging to different labels is defined by Laplace estimates.

$$P(y_i = +|e_i) = P(+|l_{i1}, \dots, l_{ik}) = \frac{n_{pos} + 1}{n_{pos} + n_{neg} + 2} \quad (1)$$

$$P(y_i = -|e_i) = P(-|l_{i1}, \dots, l_{ik}) = \frac{n_{neg} + 1}{n_{pos} + n_{neg} + 2} \quad (2)$$

where  $n_{pos}$  is the number of positive labels and  $n_{neg}$  is the number of negative labels.

In DEWMV, we calculate the voting quality of each label, which is a measure of the weight or importance of label in the integration process. Hence, the probabilistic estimate of the actual ground truth is

$$P(y_i = +|e_i) = \frac{\sum_{p=1}^{n_{pos}} w_p + 1}{\sum_{p=1}^{n_{pos}} w_p + \sum_{n=1}^{n_{neg}} w_n + 2} \quad (3)$$

$$P(y_i = -|e_i) = \frac{\sum_{n=1}^{n_{neg}} w_n + 1}{\sum_{p=1}^{n_{pos}} w_p + \sum_{n=1}^{n_{neg}} w_n + 2} \quad (4)$$

where  $w_p$  or  $w_n$  is the voting quality of label who represents the positive or negative label in the instance  $e_i$ . The  $w_{p(n)}$  does not indicate the labeler's preference for different labels or the error rate, but indicates the proportion of the label during the integration process.

Differential evolution (DE) proposed by Storn and Price [7] is a new evolutionary computation technique used to optimize real parameters or real valued functions. It is a real-coded evolutionary and heuristic algorithm based on group differences to search for the optimal solution in large spaces of solutions. In this paper, the algorithm that we proposed out based on differential evolution is achieved mainly through two steps in details: Firstly, we use differential evolution algorithm to train the all instances, with the purpose to get the adaptive

the voting quality combination of global optimum; Then, all instances can be integrated by the our algorithm under the learned global weights. The more basic strategy of our new algorithm is described as follows:

**Initialization.** For the weight individuals, we should firstly determine the population size  $N$ , and make sure that every individual in population is generated through certain mechanisms randomly.  $N$  is set to 30 in our experiment. Suppose that, the number of labels for the individual is  $k$ , a single individual population  $\mathbf{w}_i$  can be described as a  $k$ -dimensional vector:  $\mathbf{w}_i = \{w_{ij}, j = 1, \dots, k\}$  for  $i = 1, \dots, N$ , in which  $w_{ij}$  is the voting quality of label  $j$  in the individual  $i$ . The value of the voting quality for each individual is set a random number distributed between  $w_{low}$  and  $w_{up}$  ( $[0, 1]$ ).

**Mutation.** At each generation  $g$ , this operation creates mutation vectors  $\mathbf{v}_{i,g}$  based on the current parent population  $\{\mathbf{w}_{i,g} | i = 1, \dots, N\}$ . The mutate operation means that middle generations are composed with the new variation individuals from the parent generation. The following is mutation strategy frequently used in the literature and named “DE/best/2”:

$$\mathbf{v}_{i,g} = \mathbf{w}_{best,g} + F * (\mathbf{w}_{r1,g} + \mathbf{w}_{r2,g} - \mathbf{w}_{r3,g} - \mathbf{w}_{r4,g}) \tag{5}$$

where  $\mathbf{w}_{best,g}$  is the best individual in the  $g$ th generation and  $\mathbf{w}_{r1,g}$ ,  $\mathbf{w}_{r2,g}$ ,  $\mathbf{w}_{r3,g}$  and  $\mathbf{w}_{r4,g}$  which are randomly selected from the  $g$ th generation and not equal to the  $\mathbf{w}_{i,g}$  make up the difference vector used for mutate operation. And  $F$  which is a variation factor in the process of mutation can control the influence of difference vector and is set to 0.7 in experiment.

**Crossover.** The individual cross-operation in the process of crossover can increase the diversity of population and the ability of the algorithm for searching the global optimum solution. The final trial/offspring vector is  $\mathbf{u}_{i,g+1} = (u_{i1,g+1}, \dots, u_{ij,g+1})$ . The method of crossover can be described as:

$$u_{ij,g+1} = \begin{cases} v_{ij,g} & \text{if } CR \geq rand(j) \text{ or } j = randn(i) \\ w_{ij,g} & \text{if } CR < rand(j) \text{ or } j \neq randn(i) \end{cases} \tag{6}$$

where  $CR$  is the crossover probability distributed between  $[0, 1]$ , which decides the individual of replacement by mutation individual vector. And  $CR$  is equal to 0.5 in our algorithm.  $v_{ij,g}$  and  $w_{ij,g}$  denote the  $j$ th components of the mutation vector  $\mathbf{v}_{i,g}$  and the parent vector  $\mathbf{w}_{i,g}$  at generation  $g$ , respectively.  $rand(j)$  is a random number between  $[0, 1]$ ,  $j$  is the  $j$ th evaluation of a uniform random generator number.  $randn(i)$  is a randomly chosen index between  $[1, k]$  which ensures that  $\mathbf{w}_{i,g+1}$  gets at least one element from  $\mathbf{v}_{i,g}$ .

**Selection.** The selection operation selects the better one from the parent vector  $\mathbf{w}_{i,g}$  and the trial vector  $\mathbf{u}_{i,g+1}$ . If vector  $\mathbf{u}_{i,g+1}$  yields a smaller cost function value than  $\mathbf{w}_{i,g}$ , then  $\mathbf{w}_{i,g+1}$  is set to  $\mathbf{u}_{i,g+1}$ . Otherwise, the old value  $\mathbf{w}_{i,g}$  is retained. Selection operation is described as follows:

$$\mathbf{w}_{i,g+1} = \begin{cases} \mathbf{u}_{i,g+1} & \text{if } f(\mathbf{u}_{i,g+1}) < f(\mathbf{w}_{i,g}) \\ \mathbf{w}_{i,g} & \text{if } f(\mathbf{u}_{i,g+1}) \geq f(\mathbf{w}_{i,g}) \end{cases} \tag{7}$$

where the function  $f(\cdot)$  is the fitness function of DE, which isn't defined clearly in this part.

**Integration.** After the operation of iterations, we obtain the global optimal voting quality of label  $\mathbf{w}_{final}$  which is utilized in the integration process by the Eqs. (3) and (4). And then each instance in crowdsourcing would be given an integrated label.

However, we pretend that the true labels of instances are unknowable in the integration process which doesn't have training set for seeking the most perfect parameter. In our algorithm, the fitness function which is set to measure the quality of the solution obtained in DE is difficult to be defined. As a result, it is more difficult to suitably assign labels' voting weight or importance of integrated labeling. This paper attempts to use three functions as the fitness function respectively, which don't need to provide true labels of instances.

Now, the only issue left to answer is how to define the optimized fitness function. To solve this issue, we define three objective (fitness) functions: (1) the uncertainty of the integration label (simply UIL); (2) the uncertainty of the class member probability (simply UCMP); (3) the hybrid uncertainty (simply HU). And in our algorithm we minimize the fitness functions which are based on uncertainty.

**Uncertainty of the Integration Label (UIL).** The above has been introduced, MV is a simple and efficient method to integrate the true label of per instance in practice. Hence, after the step of iteration in DE, MV produces a temporary label ( $\hat{y}_e$ ) for instance  $e$  with the  $g$ th generation of the voting quality of labels. In other words, we utilize the Eqs. (3) and (4) to calculate the probability of each class and choose the maximum probability of class as the integrated label of instance. In a noisy label set of an instance, the labels which are equal to the  $\hat{y}_e$  should own the high voting quality. On the contrary, the weight of other labels would be relatively low. The fitness function of optimization is as follows:

$$f_{UIL}(\mathbf{w}_{i,g}) = \sum_{e=1}^N \sum_{j=1}^k w_{ij,g} * \mathbb{I}(l_{ej} \neq \hat{y}_e) \quad (8)$$

where  $\mathbf{w}_{i,g} = \{w_{ij,g}, j = 1, \dots, k\}$  is the  $g$ th generation of the  $i$ th population of voting weights of labels,  $l_{ej}$  is the  $j$ th noisy label for instance  $e$ , and  $\mathbb{I}(\cdot)$  is an indicator function whose output will be 1 if the test condition satisfies. Otherwise, its output will be 0.

**Uncertainty of the Class Member Probability (UCMP).** The uncertainty of the class member probability for an instance is measured by the difference between the probability that one instance belongs to a different label and the voting quality of label. The greater the difference in class member probability, the better the quality of the noisy label set. Therefore, we should reduce the uncertainties of the class member probability by constantly correcting the voting

quality of each label, and then give the following optimized fitness function:

$$f_{UCMP}(\mathbf{w}_{i,g}) = \sum_{e=1}^N 0.5 - |p(+|L_e, \mathbf{w}_{i,g}) - 0.5| \quad (9)$$

$$p(+|L_e, \mathbf{w}_{i,g}) = \frac{\sum_{j=1}^k w_{ij,g} * \mathbb{I}(l_{ej} = +) + 1}{\sum_{j=1}^k w_{ij,g} + 2} \quad (10)$$

where the  $L_e = \{l_{ej}, j = 1, \dots, k\}$  collected from crowdsourcing is the noisy label set of instance and  $l_{ej}$  is the one of labels for instance  $e$ . The  $p(+|L_e, \mathbf{w}_{i,g})$  which changes with  $\mathbf{w}_{i,g}$  represents the conditional probability that  $L_e$  and  $\mathbf{w}_{i,g}$  are provided.

**Hybrid Uncertainty (HU).** Hybrid Uncertainty is a combination of uncertainty of the integration label and uncertainty of the class member probability. And hybrid uncertainty takes full advantage of their strengths. Firstly, HU considers the influence of the majority in the integration process which shows in UIL. And then HU also includes the probability gap between the majority and minority which is disregarded in UCMP. HU is described as:

$$f_{HU}(\mathbf{w}_{i,g}) = f_{UIL}(\mathbf{w}_{i,g}) + f_{UCMP}(\mathbf{w}_{i,g}) \quad (11)$$

From the above statement, we can clearly know that the performance of HU in theory is better than UIL and UCMP, which contains the advantages of both. Thus, in our next group of experiments, HU is chosen to compare DEWMV with MV and other inference algorithms.

### 3 Experiments and Results

The purpose of this section is to validate the effectiveness of our proposed DEWMV on a collection of 14 crowdsourced problems from the main web site of the crowd environment and its knowledge analysis (CEKA) [8], which represent a wide range of domains and data characteristics. Table 1 shows the properties of these datasets.

Table 2 shows the detailed integration accuracy (%) comparisons of each algorithm on each dataset obtained on the CEKA platform. The top accuracy on each dataset is highlighted in bold. Besides, the averaged accuracies of all algorithms on 14 datasets are summarized at the bottom of the table. The average (arithmetic mean) of each algorithm across all datasets provides a gross indicator of the relative performance in addition to the other statistics. From Table 2, DEWMV obtains the top (highest) accuracies on 11 out of the 14 domains. DS, RY and PLAT obtain the top (highest) accuracies on 2, 2, and 1 domains, respectively. And the averaged accuracy of DEWMV on 14 domains is 82.23%,

**Table 1.** Descriptions of datasets used in the experiments.

Dataset	#Instances	#Labelers	#Positive	#Negative
DuchenneSmiles	159	17	642	579
Rte	800	164	4581	3419
Disgust	100	38	161	839
Duck	240	53	4309	5291
Sadness	100	38	193	807
Fear	100	38	103	897
Surprise	100	38	136	864
Joy	100	38	109	891
Trec2010	3267	722	5938	12537
Temp	462	76	2442	2178
Wordsim	30	10	142	158
Income94	600	73	5400	6599
Adult2	333	269	1211	2106
Anger	100	38	148	852

**Table 2.** Accuracy (%) comparisons for DEWMV versus MV, DS, ZC, RY, and PLAT.

Dataset	MV	DS	ZC	RY	PLAT	DEWMV
DuchenneSmiles	73.58	72.95	70.44	72.95	74.84	<b>76.73</b>
Rte	89.25	92.87	91.87	92.75	88.63	<b>92.88</b>
Disgust	79	80	79	79	79	<b>89</b>
Duck	68.3	60.8	58.75	60	76.67	<b>77.5</b>
Sadness	79	<b>83</b>	77	82	81	<b>83</b>
Fear	75	81	74	79	<b>83</b>	<b>83</b>
Surprise	53	57	52	58	66	<b>67</b>
Joy	66	75	66	<b>77</b>	74	76
Trec2010	64.37	<b>69.45</b>	58.86	67.78	64.55	66.09
Temp	94.37	94.37	94.37	94.15	93.94	<b>94.59</b>
Wordsim	90	90	86.66	86.66	90	<b>96.67</b>
Income94	73.5	72.33	73.5	72.16	71.83	<b>74.83</b>
Adult2	84.38	84.38	84.38	<b>87.98</b>	87.09	87.89
Anger	71	79	70	78	85	<b>86</b>
Average	75.77	78.01	74.06	77.67	79.68	<b>82.23</b>

**Table 3.** Summary of the Wilcoxon test.

	MV	DS	ZC	RY	PLAT	DEWMV
MV	-	○	●		○	○
DS		-	●			○
ZC	○	○	-	○	○	○
RY			●	-		○
PLAT	●		●		-	○
DEWMV	●	●	●	●	●	-

which is much higher than those of MV (75.77%), DS (78.01%), ZC (74.06%), RY(77.67%), and PLAT(79.68%), respectively.

Based on the accuracies in Table 2, we then employ the Wilcoxon signed-ranks test for thoroughly comparing each pair of algorithms. Table 3 summarizes the detailed comparison results of the Wilcoxon test. In Table 3, ○ indicates that the algorithm in the column improves the algorithm in the corresponding row, and ● indicates that the algorithm in the row improves the algorithm in the corresponding column. Lower diagonal level of significance  $\alpha = 0.05$ ; Upper diagonal level of significance  $\alpha = 0.1$ . According to the Summary of the Wilcoxon test, DEWMV performs significantly better than MV, DS, ZC, RY, and PLAT, respectively.

From these experimental results, we can see that DEWMV rarely degrades the quality of the standard MV and, in many cases, improves it remarkably. Besides, DEWMV is generally better than all the other competitors used to compare. All these comparison results indicate that searching and weighting the voting quality of each label through the designed differential evolution (DE) algorithm is highly effective for label integration.

## 4 Conclusions

Label integration continues to be a hot research topic in crowdsourcing learning, and a number of label integration methods have been proposed. In this paper we transform the label integration problem into an optimization problem, and exploit a differential evolution-based weighted majority voting method, simply DEWMV, for label integration. DEWMV searches and weights the voting quality of each label through the designed differential evolution (DE) algorithm. To define the optimized fitness function, we define three objective (fitness) functions: (1) the uncertainty of the integration label (simply UIL); (2) the uncertainty of the class member probability (simply UCMP); (3) the hybrid uncertainty (simply HU). By theoretically analysing, we choose HU as the final objective function for DEWMV. The experimental results on a collection of 14 real-world datasets show that DEWMV significantly outperforms MV and all the other state-of-the-art label integration methods used to compare.

## References

1. Dawid, A., Skene, A.: Maximum likelihood estimation of observer error-rates using the EM algorithm. *Appl. Stat.* **28**(1), 20–28 (1979)
2. Demartini, G., Difallah, D.E., Cudré-Mauroux, P.: Zencrowd: leveraging probabilistic reasoning and crowdsourcing techniques for large-scale entity linking. In: *Proceedings of the 21st International Conference on World Wide Web*, pp. 469–478. ACM (2012)
3. Howe, J.: The rise of crowdsourcing. *Wired Mag.* **14**(6), 1–4 (2006)
4. Raykar, V.C., et al.: Learning from crowds. *J. Mach. Learn. Res.* **11**, 1297–1322 (2010)
5. Sheng, V.S., Provost, F., Ipeirotis, P.G.: Get another label? Improving data quality and data mining using multiple noisy labelers. In: *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 614–622. ACM (2008)
6. Snow, R., OConnor, B., Jurafsky, D., Ng, A.Y.: Cheap and fastbut is it good? Evaluating non-expert annotations for natural language tasks. In: *Proceedings of the 2008 Conference on Empirical Method in Natural Language Processing*, pp. 254–263. Association for Computational Linguistics, Hawaii (2008)
7. Storn, R., Price, K.: Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *J. Global Optim.* **11**(4), 341–359 (1997)
8. Zhang, J., Sheng, V.S., Nicholson, B.A., Wu, X.: CEKA: a tool for mining the wisdom of crowds. *J. Mach. Learn. Res.* **16**, 2853–2858 (2015)
9. Zhang, J., Wu, X., Sheng, V.S.: Imbalanced multiple noisy labeling. *IEEE Trans. Knowl. Data Eng.* **27**(2), 489–503 (2015)





# Scalable Machine Learning Techniques for Highly Imbalanced Credit Card Fraud Detection: A Comparative Study

Rafiq Ahmed Mohammed<sup>(✉)</sup>, Kok-Wai Wong<sup>(✉)</sup>,  
Mohd Fairuz Shiratuddin, and Xuequn Wang

School of Engineering and Information Technology,  
Murdoch University, Perth, Australia

{Rafiq.mohammed, K.wong, F.shiratuddin,  
A.wang}@murdoch.edu.au

**Abstract.** In the real world of credit card fraud detection, due to a minority of fraud related transactions, has created a class imbalance problem. With the increase of transactions at massive scale, the imbalanced data is immense and has created a challenging issue on how well Machine Learning (ML) techniques can scale up to efficiently learn to detect fraud from the massive incoming data and to respond faster with high prediction accuracy and reduced misclassification costs. This paper is based on experiments that compared several popular ML techniques and investigated their suitability as a “scalable algorithm” when working with highly imbalanced massive or “Big” datasets. The experiments were conducted on two highly imbalanced datasets using Random Forest, Balanced Bagging Ensemble, and Gaussian Naïve Bayes. We observed that many detection algorithms performed well with medium-sized dataset but struggled to maintain similar predictions when it is massive.

**Keywords:** Balanced bagging ensemble · Credit card fraud detection  
Imbalanced · Machine Learning · Massive dataset · Random Forest  
Scalability

## 1 Introduction

The problem of credit card fraud detection is ‘intrinsic’ because the imbalance lies in the nature of the data space [1], and about 2% of the entire credit card transactions constitute as fraud activities [2, 3]. For this reason, the entire credit card fraud dataset becomes highly imbalanced, and having a few instances of one class means that the Machine Learning (ML) technique is often unable to generalize the behaviour of the minority class well enough. The effect of this highly imbalanced problem is ignored in the previous credit card Fraud Detection (FD) [4–6]. In addition to the complexity of imbalanced fraudulent data, the transactions data is growing massively. This is a challenging problem because in real time, we need the ML algorithm to be scalable [7] to quickly detect within two to three milliseconds fraudulent activity from massive datasets [8, 9]. Also, there are some challenges in detection such as accuracy of class prediction along with computational performance [10]. At the same time, the

unavailability of data due to privacy issues [11] is causing undersampling of fraud class [10] leading to imbalanced datasets. The objective of this paper is to perform a review of the supervised FD techniques for binary classification problem based on evaluation criteria as listed in Sect. 4.3.

## 2 Contribution

This paper is based on experiments that compared popular ML algorithms to detect credit card fraud with a massive and highly imbalanced datasets and will justify that the algorithms used are scalable enough to handle the growing amount of highly imbalanced data. This paper will also investigate scalability of the problem [7], detection algorithm adaptability with highly imbalanced data [12], their fast computation time with massive datasets [8, 9], balancing strategy [11, 12], false alarms [5], and performance metrics [11, 13]. The researchers reported in [14] evaluated the effect of hybrid sampling on performance of binary classification by comparing FD techniques for the highly imbalanced dataset, however not on the scalability perspective.

Our study is unique since it considers the scalability viewpoint of the FD techniques. In our experiments, we observed that the FD algorithms performed well in learning with patterns of data within class attributes and predicted the fraud. However, the Naïve Bayes (NB) technique struggled when we applied the generated learning model to unseen data. In addition, some techniques which are good at detecting fraud, however, lead to a lot of false alarms due to high False Positives (FP) [5]. Moreover, some techniques are good at small to medium-sized datasets, but struggle to maintain the similar accuracy of FD when the dataset is massive or big data [15]. For this reason, the scalability of the NB, Random Forest (RF), Bagging Ensemble (BE) techniques when working with massive data is taken into consideration while performing experiments, i.e. how well a FD technique responds quickly to detect fraud from the massive incoming  $t + 1$  day's data with high prediction accuracy and low false alarms. In understanding the classification technique's FD capability with newly arriving data to the Fraud Detection System, we made a change in the distribution of the data at time  $t$  and  $t + 1$ . This approach will measure the misclassification accuracy of FD with the unseen data streams.

## 3 Review of Current Credit Card Fraud Detection

Exploring FD practices based on ML technique is an appropriate approach to identify the suitable techniques for the problem domain such as credit card FD [10]. This approach will enable the researcher to understand what and why a technique works best and under which scenarios. We can classify the ML algorithms for FD into supervised and unsupervised learning techniques. The supervised learning techniques assume that labels of the past transactions, i.e. Fraud or Legitimate are available, and are reliable for fraud detection [16]. On the other hand, the unsupervised techniques do not use the class transactions labels and in the process of FD, suffer from many false alarms [17] and are more widely used in anomaly detection instead.

Learning from credit card transactions is a challenging issue, because of highly class imbalances [6]. There are different ways to handle imbalanced dataset, and we used the most commonly used Sampling and Ensemble methods in our comparison study [5]. Furthermore, while working with complex imbalanced datasets, most standard learning algorithms fail to classify FDs and often lead to high “misclassification costs” [1]. An FP rate of 30% has led to 2.7 million fraud alerts which are incorrect [18]. Class imbalance is not only the reason that deteriorates a classifier performance. Other factors such as training sample size and class complexity also influence the accuracy of FD [19].

In this study, we chose to use techniques of supervised learning for the binary classification problem because it is common for FD applications to deal with labelled data for training [8]. The supervised techniques are based on the learning and predicting model. Only few classifier techniques are available to deal with highly imbalanced credit card FD such as NB, RF, BE, Support Vector Machines (SVM), Neural Networks (NN), and k-Nearest Neighbors [6, 7]. NN classifier has some limitations as they do not work well with the highly imbalanced fraud dataset and could not generalize a good predictive model [20]. Also, NN has potential for overfitting with the training set [13].

In our experiments, SVM’s computing requirements increased rapidly with the number of training vectors, especially working with highly imbalanced massive datasets [5, 7]. NB is a scalable algorithm for highly imbalanced binary class massive datasets [15]. BE classifier based on decision trees is proven to be suitable when working with highly imbalanced massive credit card fraud transactions datasets [7]. RF has the ease of implementation, high classification accuracy with the imbalanced datasets [21], fast computation time [8] and is scalable to work with a massive datasets [8, 22]. Having to consider the classifier scalability to detect fraud with the highly imbalanced and massive datasets in near real time environment, we have selected NB, BE and RF.

## 4 Experiments

In this paper, we conducted multiple experiments on two datasets. We sourced the first dataset from the European Credit Card (ECC) transactions provided by the ULB ML Group [23]. This dataset contains anonymized 284,807 highly imbalanced credit card transactions with an Imbalance Ratio of 1:578, and 0.17% or 492 of fraudulent transactions. We sourced the second dataset from the Revolution Analytics [24]; the dataset contains 10 million credit card transactions, which is massive with imbalanced ratio of 1:16, and consists of 5.96% of fraudulent transactions.

### 4.1 Dealing with Imbalanced Data

In a class imbalance problem, the techniques to handle imbalanced dataset are applied either at the data level or an algorithmic level [5]. Similarly, we can classify the sampling methods into under-sampling and over-sampling. A common strategy is to under-sample the majority class in the training set before learning a classifier [25].

Moreover, we can over-sample the minority class in the training set before learning a classifier. The latest methods such as the ‘Synthetic Minority Over Sampling Technique’ (SMOTE) [26] is available to improve the balancing strategy in the binary classification problem. Another method known as the ‘Ensemble’ method combines balancing techniques with a classifier to explore the majority and minority class distribution [11]. Additionally, it is categorized as cost-sensitive method for class imbalance problem [27]. Literature suggests that the best approach to handle imbalanced datasets does not exist [28]. However, it is possible to adjust a sampling strategy dynamically by the process of active learning [28]. To achieve the objective of this paper to review the supervised FD techniques, we explored different sampling techniques and rates.

## 4.2 Performance Metrics

Measuring the success of ML algorithms [10] is an important task, so that the best algorithm suitable for the problem such as credit card FD can be selected [13]. We have used ‘*F-measure*’ which is Harmonic Mean (HM) of precision and recall is more appropriate for the successful separation between fraud and legitimate transactions [29]. Additionally, we have used Area Under Curve (AUC) Receiver Operating Characteristic (ROC) “predict probability” instead of predict class for AUC measure [11, 30].

## 4.3 Experimental Setup

In this section, we list the evaluation criteria of the classifier. For the experimental setup, we analyzed approaches on selecting best sampling technique and set the optimized parameters of the techniques. We ran experiments using Python 3.5 software.

### Classifier evaluation criteria

As articulated earlier in the contributions section, when building a classifier model of fraud detection, the following impact factors were considered to be effective.

*Scalable [8] with massive and highly imbalanced datasets [7, 12, 31] and detect frauds accurately [6]. Faster computation time with massive datasets [8, 9, 22] for early fraud detection. Has low false alarms [5, 31] and low fraud misclassifications [6, 27], and maintain similar prediction rates with the unseen data [8, 11].*

### Balancing strategy

In resolving a highly class imbalance problem, we experimented the balancing strategy available for datasets. We explored sampling distribution option when we analyzed the datasets by taking into consideration our classifier evaluation criteria. Since one of our impact factors for the model evaluation criteria is that a classifier technique should maintain similar prediction rates with the unseen data [11], in validating the effectiveness of the model, a 10% of the data has been set aside as dataset1 for  $t + 1$  day’s test data. Dataset2 which has 90% of the data was used to train the model, validate and

test the model predictions. Since the dataset is highly imbalanced, a 10:90 sampling rate selection on dataset2 has achieved comparatively superior results of predictions.

### Parameters selection

The classifier model's parameters selection has been optimized using Grid search CV [32] functionality available in Python software package and based on this functionality, tuning the best parameters list for the classifier model has been selected.

## 4.4 Results and Discussions

In this study, we used three classifier models; the RF, BBE, and GNB. We have applied various balancing techniques such as the Random Under Sampling (RUS), Random Over Sampling (ROS), various flavours of SMOTE (original, borderline1, borderline2, SVM), SMOTEENN, and SMOTETomek to both datasets, and evaluated their performance. In this paper, we presented the three best combinations of balancing technique and learning classifier, based on the performance metrics. Table 1 shows results of the experiments for FD using the ECC dataset, and Table 2 shows the results using ccFraud dataset. We have summarized the analyses of the results in Table 3.

**Table 1.** Accuracy result for European credit card dataset.

Metrics	BBE	SMOTE (bl1) + BBE	SMOTE (bl2) + BBE	RUS + GNB	SMOTE + GNB	SMOTE (bl2) + GNB	RUS + RF	SMOTE (bl1) + RF	SMOTE (bl2) + RF
Sensitivity	<b>0.9388</b>	0.8990	0.8776	0.8163	0.8571	0.6735	0.8776	0.8980	0.8980
Specificity	0.9737	<b>0.9998</b>	<b>0.9998</b>	0.9862	0.9925	0.9953	0.9983	0.9994	0.9991
Precision	0.0641	<b>0.9167</b>	0.8776	0.1020	0.1795	0.2143	0.5000	0.7458	0.6567
Avg precision recall	0.06	<b>0.82</b>	0.77	0.08	0.15	0.14	0.44	0.67	0.59
AUC ROC	<b>0.9801</b>	0.9584	0.9582	0.9560	0.9604	0.9638	0.9536	0.9702	0.9688
F1 Score	0.1199	<b>0.9072</b>	0.8776	0.1814	0.2968	0.3251	0.6370	0.8148	0.7586
<u>Blind Tests</u>									
Sensitivity	<b>0.7727</b>	0.6818	0.7273	0.5000	0.5000	0.5000	0.7273	0.7273	0.7273
Specificity	0.9721	0.9999	<b>1.0000</b>	0.9887	0.9928	0.9955	0.9991	0.9993	0.9986
Precision	0.0210	0.8824	<b>0.9412</b>	0.0331	0.0509	0.0786	0.3721	0.4324	0.2807
Avg precision recall	0.02	0.60	<b>0.68</b>	0.02	0.03	0.04	0.27	0.31	0.20
False positive rate	2.79%	0.01%	<b>0%</b>	1.13%	0.72%	0.45%	0.09%	0.07%	0.14%
AUC ROC	<b>0.9811</b>	0.9064	0.8836	0.9695	0.9724	0.9548	0.9508	0.9614	0.9712
F1 Score	0.0409	0.7692	<b>0.8205</b>	0.0621	0.0924	0.1358	0.4923	0.5424	0.4051
Runtime (in seconds)	147.60	367.81	368.25	<b>141.03</b>	145.44	145.27	143.72	535.51	563.61

**Table 2.** Accuracy result for the ccFraud dataset.

Metrics	BBE	ROS + BBE	SMOTE (bl1) + BBE	RUS + GNB	SMOTE + GNB	SMOTE (bl1) + GNB	RUS + RF	ROS + RF	SMOTE (bl1) + RF
Sensitivity	0.8532	0.6428	0.3921	0.8499	0.8603	0.9023	<b>0.9113</b>	0.9068	0.8530
Specificity	0.8588	0.9382	<b>0.9767</b>	0.8805	0.8750	0.8408	0.8661	0.8733	0.9068
Precision	0.2780	0.3985	<b>0.5176</b>	0.3119	0.3048	0.2653	0.3026	0.3132	0.3685
Avg precision recall	0.25	0.28	0.24	0.27	0.27	0.25	0.28	0.29	<b>0.32</b>
AUC ROC	0.9536	0.9226	0.8995	0.9391	0.9400	0.9408	0.9573	<b>0.9583</b>	0.9512
F1 score	0.4194	0.4920	0.4462	0.4563	0.4502	0.4100	0.4543	0.4656	<b>0.5146</b>
Blind Tests									
Sensitivity	0.8497	0.6380	0.3883	0.8485	0.8588	0.9020	<b>0.9098</b>	0.9037	0.8500
Specificity	0.8584	0.9376	<b>0.9766</b>	0.8796	0.8740	0.8402	0.8656	0.8727	0.9063
Precision	0.2740	0.3913	<b>0.5104</b>	0.3072	0.3001	0.2602	0.2986	0.3088	0.3634
Avg precision recall	0.24	0.27	0.23	0.27	0.27	0.24	0.28	0.28	<b>0.32</b>
False positive rate	14.2%	6.2%	<b>2.3%</b>	12.0%	12.6%	16.0%	13.4%	12.7%	9.4%
AUC ROC	0.9346	0.9218	0.8980	0.9383	0.9392	0.9399	0.9564	<b>0.9573</b>	0.9502
F1 score	0.4144	0.4851	0.4410	0.4511	0.4448	0.4061	0.4496	0.4603	<b>0.5091</b>
Runtime (in seconds)	90	1032	33264	<b>21</b>	61	28501	147	3103	34086

**Table 3.** Analyses of the results of the experiments.

Evaluation criteria	Balanced bagging ensemble	Gaussian naïve bayes	Random Forest
Scalable with highly imbalanced massive datasets and detects frauds accurately	Fraud catching rate of 85% with massive datasets	Second best of 90% recall after RF for massive datasets	A good prediction of 90% with the ECC. Best fraud prediction of 91% with massive datasets
Faster computation time with massive datasets for early fraud detection	Faster with smaller datasets. Faster using RUS for massive datasets	Fastest with smaller datasets. Faster using RUS and ROS for massive datasets	RF using RUS was able to detect fraud in just over 2 min
Low false alarms and low fraud misclassifications	‘Nil’ false alarms with the ECC. Improving precision drop FD rate with massive datasets	Comparatively high false positive rates leading to false alarms	Second highest false positive rate. However, there is a trade-off with good recall
Similar predictions with unseen data	Superior prediction with the ECC using hybrid ensemble (RUS & SMOTE - borderline2)	Maintained similar predictions with massive dataset only	Good predictions with massive datasets

Based on our analyses, the **Gaussian Naïve Bayes** technique is comparatively faster than the RF and BBE classifier in detecting fraud. It took just over 2 min for ECC dataset. Based on SMOTE, it has achieved a Recall of 86% and an F1 score of 0.30 and AUC ROC of 0.96. However, the precision rate of 18% is too low which leads to many false alarms. When we tested the generated model on unseen  $t + 1$  days of data, it could not generate similar prediction rates as of 't' days of data. The FD rate of only 50% due to the false alarms and FP rate of 0.72% and the achieved average precision is only 3%.

The next set of experiments were conducted using ccFraud dataset of 10 Million records. GNB with SMOTE could detect 86% of fraud but at the expense of precision of 30%. In improving the recall, we applied SMOTE (borderline1) in which we gained on FD of 90%. However, at a loss of precision, which dropped to 26% and expensive computational time of 8 h. Overall, when we tested the GNB model on unseen  $t + 1$  days of data, it could generate similar prediction rates as of 't' days of data. However, the overall GNB has the highest FP rate of 16%, and for this reason, it could neither improve its precision nor the F1 score. Literature suggests that when considering the FP rate of 13%, NB detected 650 k of fraud alerts as per 2013 industry standards [18].

The **Balanced Bagging Ensemble** (BBE) which is balanced internally with the RUS technique [33] achieved a FD rate of 94% with the ECC dataset. However, the average precision of 6% was too low. To improve, we used hybrid ensemble (RUS by default + SMOTE (bl1)) and achieved a FD rate of 90%, and a precision of 92% and AUC of 0.96, and an F1 score of 0.91. The ensemble with RUS + SMOTE (bl2) achieved 'nil' false alarms and could able to generate a recall of 73% with a precision of 94%, and an F1 score of 0.82 and AUC of 0.88 with unseen  $t + 1$  days of FD.

When we conducted experiments using massive dataset, the BBE was able to achieve a FD rate of 85% but with a precision of 28%. In improving precision, we applied the hybrid sampling methods. As evidenced in Table 2, the 12% improvement of precision with ROS hybrid ensemble led to more than 20% drop in FD and achieved a rate of 65%. When we tried the hybrid ensemble with SMOTE (bl1), we were able to double the precision to 52%. However, the FD rate dropped to less than half to 40%. Surprisingly enough, the FP rate dropped drastically with the gain of precision which was a good sign. On a bright side, the BBE when applied to the massive dataset, could maintain similar prediction rates with the unseen  $t + 1$  days of data.

The **Random Forest (RF) Classifier** technique when working with the ECC dataset was able to detect fraud at 90% with AUC of 0.97 and an F1 score of 0.81. For the  $t + 1$  days of FD, the RF model based on SMOTE (bl1) was able to generate FD of 73%, with a precision of 43%, an F1 score of 0.54 and AUC of 0.96. Overall, when considering the model predictions for  $t + 1$  days, the RF ranked second best next to the BBE.

The RF classifier proved to be a good classifier when working with the massive dataset. The experiments showed that both the RUS & ROS methods for RF using blind test, gave a stable FD with upcoming new data. We have achieved similar prediction rates of 91% and 90% with AUC of 0.96. However, the RUS has accumulated 6,731 additional FPs as compared to ROS. Furthermore, RUS could detect 356 additional fraudulent transactions compared to ROS. In reality, when we want to achieve higher recall, it will be at the expense of lower precision [34] and vice-versa.

Since RF using RUS was able to detect fraud in just over 2 min with the highest recall of 91%, this technique would be of high value working in the real time environment.

## 5 Conclusion

Real time credit card fraud detection is a challenging issue due to highly imbalanced massive data. This research paper is based on experiments that compared several popular ML techniques and investigated their suitability as a “scalable algorithm” when working with highly imbalanced massive or “Big” datasets. In summary, when considering fraud detections for ‘t’ and t + 1 days, Balanced bagging ensemble with Hybrid (RUS & SMOTE-borderline2) balancing technique has superior prediction with ECC dataset. Random Forest with RUS is proven to be scalable and capable of FD with highly imbalanced massive datasets. Future directions will be based on developing scalable Fraud Detection System in Big Data environment for highly imbalanced datasets.

## References

1. He, H., Garcia, E.A.: Learning from imbalanced data. *IEEE Trans. Knowl. Data Eng.* **21**(9), 1263–1284 (2009)
2. Japkowicz, N., Stephen, S.: The class imbalance problem: a systematic study. *Intell. Data Anal.* **6**(5), 429–449 (2002)
3. Juszczak, P., et al.: Off-the-peg and bespoke classifiers for fraud detection. *Comput. Stat. Data Anal.* **52**(9), 4521–4532 (2008)
4. Dal Pozzolo, A., Caelen, O., Bontempi, G.: When is undersampling effective in unbalanced classification tasks? In: Appice, A., Rodrigues, P.P., Santos Costa, V., Soares, C., Gama, J., Jorge, A. (eds.) *ECML PKDD 2015. LNCS (LNAI)*, vol. 9284, pp. 200–215. Springer, Cham (2015). [https://doi.org/10.1007/978-3-319-23528-8\\_13](https://doi.org/10.1007/978-3-319-23528-8_13)
5. Ali, A., Shamsuddin, S.M., Ralescu, A.L.: Classification with class imbalance problem: a review. *Int. J. Adv. Soft Comput. Appl.* **7**(3), 176–204 (2015)
6. Zareapoor, M., Yang, J.: A novel strategy for mining highly imbalanced data in credit card transactions. *Intell. Autom. Soft Comput.* 1–7 (2017). <https://doi.org/10.1080/10798587.2017.1321228>, ISSN 1079-8587
7. Zareapoor, M., Shamsolmoali, P.: Application of credit card fraud detection: based on bagging ensemble classifier. *Procedia Comput. Sci.* **48**, 679–685 (2015)
8. Carneiro, N., Figueira, G., Costa, M.: A data mining based system for credit-card fraud detection in e-tail. *Decis. Support Syst.* **95**, 91–101 (2017)
9. PYMNTS Homepage. AI Puts Fraudulent Credit Card Testers To The Test, 21 February 2018. <https://www.pymnts.com/fraud-prevention/2018/brighterion-credit-card-fraud-prevention/>. Accessed 24 Mar 2018
10. West, J., Bhattacharya, M.: Intelligent financial fraud detection: a comprehensive review. *Comput. Secur.* **57**, 47–66 (2016)
11. Dal Pozzolo, A., et al.: Learned lessons in credit card fraud detection from a practitioner perspective. *Expert Syst. Appl.* **41**(10), 4915–4928 (2014)



12. Lu, Y., Cheung, Y.-m., Tang, Y.Y.: Hybrid sampling with bagging for class imbalance learning. In: Bailey, J., Khan, L., Washio, T., Dobbie, G., Huang, J.Z., Wang, R. (eds.) PAKDD 2016. LNCS (LNAI), vol. 9651, pp. 14–26. Springer, Cham (2016). [https://doi.org/10.1007/978-3-319-31753-3\\_2](https://doi.org/10.1007/978-3-319-31753-3_2)
13. West, J., Bhattacharya, M.: Some experimental issues in financial fraud mining. *Procedia Comput. Sci.* **80**, 1734–1744 (2016)
14. Awoyemi, J.O., Adetunmbi, A.O., Oluwadare, S.A.: Credit card fraud detection using machine learning techniques: a comparative analysis. In: 2017 International Conference on Computing Networking and Informatics (ICCN). IEEE (2017)
15. Liu, B., et al.: Scalable sentiment classification for big data analysis using Naive Bayes Classifier. In: 2013 IEEE International Conference on Big Data. IEEE (2013)
16. Bolton, R.J., Hand, D.J.: Statistical fraud detection: a review. *Stat. Sci.* **17**, 235–249 (2002)
17. Dai, Y., et al.: Online credit card fraud detection: a hybrid framework with big data technologies. In: Trustcom/BigDataSE/I SPA, 2016 IEEE. IEEE (2016)
18. Ryman-Tubb, N.: Understanding payment card fraud through knowledge extraction from neural networks using large-scale datasets. University of Surrey (2016)
19. Japkowicz, N.: Class imbalances: are we focusing on the right issue. In: Workshop on Learning from Imbalanced Data Sets II (2003)
20. 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](https://doi.org/10.1007/978-981-4585-18-7_2)
21. Ma, L., Fan, S.: CURE-SMOTE algorithm and hybrid algorithm for feature selection and parameter optimization based on random forests. *BMC Bioinf.* **18**(1), 169 (2017)
22. Han, J., Liu, Y., Sun, X.: A scalable random forest algorithm based on mapreduce. In: 2013 4th IEEE International Conference on Software Engineering and Service Science (ICSESS). IEEE (2013)
23. European Credit Card dataset. U.M.L. Group, Editor, ULB Machine Learning Group (2013). <https://www.kaggle.com/mlg-ulb/creditcardfraud>
24. ccFraud dataset, April 2013. <https://packages.revolutionanalytics.com/datasets/>
25. Akbani, R., Kwek, S., Japkowicz, N.: Applying support vector machines to imbalanced datasets. In: Boulicaut, J.-F., Esposito, F., Giannotti, F., Pedreschi, D. (eds.) ECML 2004. LNCS (LNAI), vol. 3201, pp. 39–50. Springer, Heidelberg (2004). [https://doi.org/10.1007/978-3-540-30115-8\\_7](https://doi.org/10.1007/978-3-540-30115-8_7)
26. Chawla, N.V., et al.: SMOTE: synthetic minority over-sampling technique. *J. Artif. Intell. Res.* **16**, 321–357 (2002)
27. Galar, M., et al.: A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches. *IEEE Trans. Syst. Man Cybernet. Part C (Appl. Rev.)* **42**(4), 463–484 (2012)
28. Provost, F.: Machine learning from imbalanced data sets 101. In: Proceedings of the AAAI 2000 Workshop on Imbalanced Data Sets (2000)
29. Fisher, W.D.: Machine Learning for the Automatic Detection of Anomalous Events. ProQuest Dissertations Publishing (2017)
30. Géron, A.: Hands-on Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. O'Reilly Media Inc., Sebastopol (2017)
31. Carcillo, F., et al.: An assessment of streaming active learning strategies for real-life credit card fraud detection. In: 2017 IEEE International Conference on Data Science and Advanced Analytics (DSAA). IEEE (2017)

32. Pedregosa, F., et al.: Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* **12**(Oct), 2825–2830 (2011)
33. Lemaitre, G., Nogueira, F., Oliveira, D., Aridas, C.: *BalancedBaggingClassifier* (2016). <http://contrib.scikit-learn.org/imbalanced-learn/stable/generated/imblearn.ensemble.BalancedBaggingClassifier.html>. Accessed 17 Mar 2018
34. Saito, T., Rehmsmeier, M.: The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PLoS ONE* **10**(3), e0118432 (2015)



# View Decomposition and Adversarial for Semantic Segmentation

He Guan<sup>1,2,4</sup> and Zhaoxiang Zhang<sup>1,2,3,4</sup>(✉)

<sup>1</sup> University of Chinese Academy of Sciences, Beijing, China  
{guanhe2015,zhaoxiang.zhang}@ia.ac.cn

<sup>2</sup> Research Center for Brain-inspired Intelligence, CASIA, Beijing, China

<sup>3</sup> CAS Center for Excellence in Brain Science and Intelligence Technology,  
Beijing, China

<sup>4</sup> National Laboratory of Pattern Recognition, CASIA, Beijing, China  
tnt@nlpr.ia.ac.cn

**Abstract.** The adversarial training strategy has been effectively validated because it maintains high-level contextual consistency. However, limited to the weak capability of a simple discriminator, it is irresponsible and unreasonable to identify one from the sample source at a time. We introduce a novel discriminator module called Multi-View Decomposition which transforms the discriminator role from general teacher to specific adversary. The proposed module separates single sample into a series of class inter-independent streams and extracts corresponding features from current mask. The key insight in the MVD module is that the final source decision can be aggregated from all available views rather than a harsh critic. Our experimental results demonstrate that the proposed module can improve performance on PASCAL VOC 2012 and PASCAL Context dataset further.

**Keywords:** View decomposition · Adversarial  
Semantic segmentation

## 1 First Section

### 1.1 Introduction

Semantic segmentation is a fundamental computer vision problem where the goal is to assigns all pixels into different semantic classes. It enjoys a wide range of applications such as self-driving systems and scene parsing. There have been some recent efforts on adapting fully convolutional network for semantic segmentation and achieved state-of-the-art performance. Some of these work proposed various exclusive network architectures to embed higher-level convolutional features from multiple layers in CNN. Others are based on a variety of post-processing methods by integrating higher-order potentials as much as possible to enforce spatial contiguity in score maps. In both cases, the final semantic

predictions attempt to overcome the limitation of the barrel-shaped network architecture and enhance the correlation among neighboring nodes.

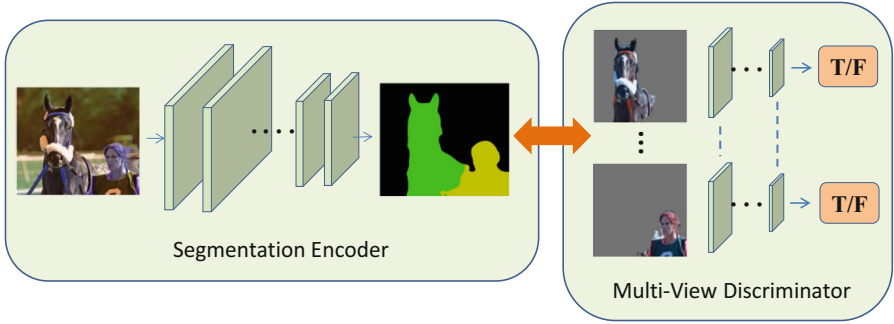
Generative Adversarial Network is a powerful approach to identify the authenticity of sample. We expect the desirable state if a generated sample can be confused with a genuine sample. In the process of alternating iterations, discriminator tends to converge rapidly while generator is hard to update. On the contrary, it reverses the dilemma when the generator is initialized with a well pre-trained segmentation model [12]. With the enhancement of model robustness and generalization, the appearance gap between real sample and synthetic ones is gradually disappearing and the discriminator will quickly obtain the local optimal solution. Compared with the standard GAN, the gradients of feedback are less of deliberation and more like random allocation in this state.

Based on above observation, we hypothesis that multi-discriminator variants can collect more specific representations from different subspace and approximate  $\max_D V(D, G)$  better by ensemble learning and propose a novel discriminator to deal with the above shortcoming. Different from previous work, we make a series of class inter-independent streams for each category and aggregate them into a more harsh critic via global optimization. The final paradigm is a pattern of one generator against one discriminator with multi-view. Even if the semantic segmentation model produce relatively realistic labeling decision, it can still exploit the potential differences in Contextual details and avoid misleading guidance. Extensive experiments on two benchmark datasets demonstrate the superiority of our Multi-View Decomposition module.

## 2 Related Work

Following Long [11] replaced fully-connected layers in classification into convolutional layers, various FCN architectures have made breakthroughs constantly in semantic segmentation task. To make use of global prior from complex objects and scenes, some methods [9, 18] extract Context information using global pooling branches while others embed or combined the feature map from multi-scale [15, 17] and multi-resolution [2, 8].

Recently, Various GAN variants are not satisfied with the structural limitations of one generator against one discriminator. Nguyen et al. propose D2GAN model [14] with different rewards using the complementary statistical properties of Kullback-Leibler divergence and reverse-KL divergence for single discriminator against double discriminators. Both the Multi-View GAN [4] and GMAN [5] extend to the universal paradigm of single generator against multiple discriminators. As for apply adversarial training to semantic segmentation task, Luc et al. [12] take the lead to combine both and prove its effectiveness by optimize a general multi class cross-entropy loss with an additional adversarial term. Natalia et al. [13] expand his work on predicting semantic segmentation maps in future video frames. Souly et al. [16] replace the discriminator with baseline segmentation model rather than generator to leverage from generated output or unlabeled data. These work shows that GAN framework can flexibly extend to semantic segmentation task.



**Fig. 1.** Overview of the adversarial training using Multi-View Decomposition module. Given an input image, we first feed it to segmentation generator to product a dense prediction map, then the MVD module is applied to extract independent features from one-hot encoding mask on image, followed by concatenation layers and the rest of convolution layers to form our discriminator. (Color figure online)

The purpose of GANs above are to enhance the robustness and generalization for the generator and obtain enough realistic samples to deceive the discriminator. Due to the situation of image blurring, deformation and distortion, it is obviously easy for simple classifier to refuse these fake case. The difference is that the rough segments discard all high-frequency details except the structural relationship. It is not representative enough to pick out the fakes once the generation results has a rough prototype. Therefore, we exploit the capability of discrepancy by multiple discriminators collaborate via our Multi-View Decomposition module.

### 3 Approach

In this section, we propose a effective module as feature extractors called the Multi-View Decomposition as the discriminator to discern faint differences. An overview of the MVD module is shown in Fig. 1. Our method has few differences from advances in several architectural choices for the discriminator such as convolutional Patch-GAN [7]. Unlike past work, the first few layers of MVD are replaced to penalizes large portions of image patches. Here we show that the multi-view learning method captures the characteristics of a particular category through a dedicated hierarchical structure. The timing of separation and fusion is explored in Sect. 4.3.

The MVD module is motivated by the following observations: The structure of semantic graph only contains spatial continuity information about the objects without enough precise texture details. It is no significant difference when one-hot coding of the ground truth ones or the probability maps as the input no matter whether the corresponding RGB images or not. Part of the explanations for this low gain is that the discrete or continuous maps have no enough driving force for discriminator to distinguish them. However, the essential aim of introducing adversarial training is to promote the segmentation network output

more realistic and reasonable. Thus, it should not be limited its potential by the trivial judgment.

The discriminator of our network has eight convolution layers and we split front-end network as the MVD before the  $k$ -th convolutional layer ( $k = 2, 3, 4$ ). Excessive proportion of this module will leads to some signal dominates the others if propagate to subsequent layers. It is also necessary to extract features by hierarchical branches according to categories because the next following layer will mix the inter-class features immediately, which is equal to directly input the original RGB images or label maps. Weights sharing is one effective method for learning approximate features and greatly increases the versatility of the filters. We expect MVD to be more sensitive in local differences than stacking multiple convolution blocks by single stream, so the weights of different branches are not shared. The cost is we need to slightly increase the scanning frequency for one image. We believe it is more likely to train alternately from distinct views of the sample and jointly optimize the others to maximize the consensus in a series of subspace.

We assign a multi-class cross-entropy loss  $L_{mce}$  and a auxiliary adversarial regularization loss such as  $L_1$  to approximate between the model output  $\hat{Y}$  with ground truths  $Y$ . An additional weight is responsible for balancing the influence of auxiliary losses. We provides the optimal balance value using  $\lambda = 0.5$ .

$$\mathcal{L}_G(\hat{Y}, Y) = \mathcal{L}_{mce}(\hat{Y}, Y) + \lambda \cdot \mathcal{L}_{L1}(D(\hat{Y}), \alpha) \quad (1)$$

$$\mathcal{L}_D(\hat{Y}, Y) = \mathcal{L}_{L1}(D(Y), \alpha) + \mathcal{L}_{L1}(D(\hat{Y}), 0) \quad (2)$$

Given the number of annotations class  $C$  and input RGB image size  $H \times W$ , we minimize the  $\mathcal{L}_G$  to produce more precise segmentation maps to fool the adversarial model by generative part, while the  $\mathcal{L}_D$  is trained to ferret out the attacker from the pairs of feeding samples by the discriminator  $D$ . Using  $|\cdot|$  denotes the absolute value function following with spatial position  $i, j$  and class index  $c$ . In addition, one-side label smoothing technique [1] is used to reduce the vulnerability of the neural network to counterattacks. We restrict positive cases with  $\alpha = 0.9$  and negative cases with  $\beta = 0$  because moderately positive sample smoothing is helpful to obtain stronger gradient feedback.

Since convolution with stride change can maintain more details than max pooling, we implement down-sampling operation by stride convolution layer rather than max-pooling layer. Convolution-ReLU form is also used to most layer except the first and final one. After eight convolution layers with  $3 \times 3$  spatial filters, the fields-of-view has reached  $34 \times 34$  pixels at the top of the discriminator which is similar to LargeFOV [12]. Note that current state-of-the-art segmentation network generally shrink eight times as output scale and zoom back to original size during test [3, 17, 18]. We do not expand the receptive field deliberately in the MVD further since it is sufficient to detect the sharpness of class boundaries and avoid tiling artifacts. Beyond this scale, to cover larger image patch, will bring considerably worse results. This may be because this front-end has many more parameters than before in the same depth of the architecture, which may be harder to converge.

## 4 Experiments

In this section, we first briefly describe the implementation details and the baseline model. Then we evaluate our method on two standard benchmarks: PASCAL VOC 2012 dataset and PASCAL Context dataset. In the end, we give a contrast verification on the impact of the module fusion depth and visual results analysis.

### 4.1 Implementation Details and Baselines

All experiments are done on the open source framework **Tensorflow**. The training optimizer selects Adam for generator and SGD for discriminator. Moreover, the learning rate is set to 0.0001 and the momentum remains 0.9. For the PASCAL VOC 2012 and PASCAL Context dataset, we randomly crop a  $321 \times 321$  region from each image during training and substrate image pixel mean to normalize at every position. In contrast, we generate a broader coverage map of  $512 \times 512$  then crop it to remove useless areas during test.

We have tried three alternative updating strategies to explore the effect of the switching speeds between  $G$  and  $D$ . The fast version implements a 1 : 1 switching rate, while the constant version prefers 20 iterations intervals and the last one corresponds to slow frequency version such as 1 : 500. Through verification, we found that switching from the slow to constant mode during training phase can help accelerate the network convergence.

We only focus on the extra gain by using our novel module in adversarial training so that the generator can be transferred into any semantic segmentation baseline model directly. Here we employ the state-of-the-art semantic segmentation network of [17] or [3]. One concern is that the former can compare with [12] and the latter can apply more pressure on the discriminator to test its strength. Both [3,17] are based on the VGG-16 pre-trained model and apply several dilation convolution layers as substitutes for max-pooling. Additional Context module or multi-scale convolution branches can capture both short and long range contexts by expand its receptive fields.

The rest of the network parameters are initialized by COCO data pre-trained, which can suppress the interference caused by the cold start of the discriminator. Then we turn on branches alternately to model each particular class and follow the co-training strategy in multi-view learning to maximize the mutual agreement, which is similar to fine-tune current path only on current class information. More frequent alternating scheme is more applicable for solving the class imbalance problem.

### 4.2 Datasets

PASCAL VOC 2012 is a generic segmentation benchmark, which contains 20 categories and the background. Following common practice [3,10,11], we use augmented data with extra SBD [6] resulting 10582 images for training. We also respectively validate and test on the original 1449 and 1456 images. PASCAL Context dataset contains 10103 images with 540-classes dense label,

**Table 1.** Comparison results on the PASCAL Context using DeepLab-VGG. The preceding symbol <sup>†</sup> indicates fine-tuned on PASCAL Context dataset using original environment configuration.

Method	mIoU	Pixel Acc.
DeepLab-v2 <sup>†</sup>	41.0	63.7
Ours (with BCE)	41.1	63.7
Ours (with L1)	<b>41.5</b>	<b>64.4</b>

**Table 2.** Comparison results on different dataset for fusion depth.

Fusion Depth	1st	2nd	3rd	4th	Avg
Pascal VOC mIoU	71.94	72.31	<b>72.86</b>	70.09	72.01
Pascal Context mIoU	41.0	41.2	<b>41.5</b>	40.4	40.9

which is split to 4998 images for training and 5015 images for validation. We only consider the most frequent 60 classes (including background) for training and evaluation, regardless of those low-frequency object. As for evaluation, both mean intersection over union (mIoU) and pixel-wise accuracy (Pixel Acc.) are used. More results are shown in Tables 1 and 3.

### 4.3 Fusion Strategy

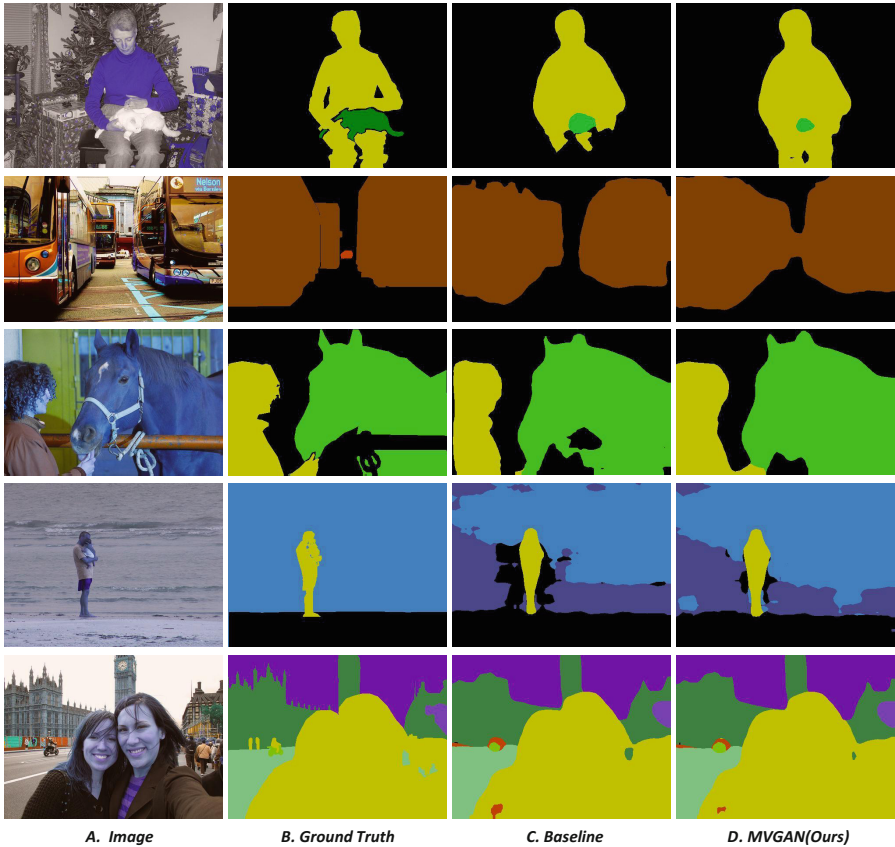
We compare the adversarial training results by modifying the module depth in discriminator, so as to seek out the best-fit fusion depth in variations. The results are showed in Table 2. It is obvious that stacking more layers gives slight improvements over the shallower fusion mode in contrast with a negative effect on network once beyond the upper limit. We believe that excessive proportion of independent channels will leads to some signal dominates the others if propagate to subsequent layers. Besides, we further evaluate an additional framework that the final confidence score map is averaged by multiple discriminant outputs (*Avg*) at each position, which become a boosting algorithm in this extreme case. The module depth is eventually set to 3 to trade off the independent single-view feature extraction process and the multi-view feature integration process. Note that it doesn't matter about the receptive field scale, regardless of the merge position.

### 4.4 Visual Analysis

Here we analyze several visual examples shown in Fig. 2, and demonstrate that multi-view decomposition and adversarial strategy can well cope with such problems as insufficient coherence of semantics and the misclassification of the pixels inside objects. As we observed that the completed human torso under the sleep



cat (1st); or the correction of a wide range of sea background semantics (4th). However, there are still some samples that are corrected only in weak areas, especially when the fake samples were highly smooth such as shaded bus (2nd) or coupled girl bodies (5th).



**Fig. 2.** Visual quality comparison results on different datasets.

**Table 3.** Per-class results on the PASCAL VOC 2012 test set

Method	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mIoU
Dilation-8	87.2	38.2	84.5	62.3	69.7	88.0	82.3	86.4	34.6	80.5	60.5	81.0	86.3	83.0	82.7	53.6	83.9	54.5	79.3	63.7	73.1
SSGAN	87.1	38.5	84.9	63.2	69.7	88.0	82.5	86.8	34.5	80.3	61.5	80.9	85.8	83.3	82.6	55.0	83.5	54.7	79.7	62.9	73.3
Ours	87.2	38.7	84.8	63.5	69.8	88.4	82.5	87.1	34.7	80.4	61.7	81.3	86.2	83.4	83.3	54.8	83.8	54.6	79.9	63.8	73.7

## 5 Conclusion

Following the principle of multi-view learning, we propose a new module for adversarial training by update particular features alternately on split segmentation maps. It concentrates more on intra-class features and exploits the redundant views of the same input data to optimize the discriminant state without reducing the regularization property of higher-order statistics too much. Our results demonstrate that our proposed module enhance the discriminator in semantic segmentation task and still can improve the performance on several datasets even if the segment network enough powerful. For future work, we plan to explore multi-view adversarial training on different attributes of homologous data such as age, gender or expression of human face.

**Acknowledgement.** This work was supported in part by the National Key R & D Program of China(No. 2018YFB1004600), the National Natural Science Foundation of China (No. 61773375, No. 61375036, No. 61602481, No. 61702510), and in part by the Microsoft Collaborative Research Project.

## References

1. Arjovsky, M., Bottou, L.: Towards principled methods for training generative adversarial networks. arXiv preprint [arXiv:1701.04862](https://arxiv.org/abs/1701.04862) (2017)
2. Bansal, A., Chen, X., Russell, B., Ramanan, A.G., et al.: Pixelnet: Representation of the pixels, by the pixels, and for the pixels. arXiv preprint [arXiv:1702.06506](https://arxiv.org/abs/1702.06506) (2017)
3. Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. PAMI (2018)
4. Chen, M., Denoyer, L.: Multi-view generative adversarial networks. In: Ceci, M., Hollmén, J., Todorovski, L., Vens, C., Džeroski, S. (eds.) ECML PKDD 2017. LNCS (LNAI), vol. 10535, pp. 175–188. Springer, Cham (2017). [https://doi.org/10.1007/978-3-319-71246-8\\_11](https://doi.org/10.1007/978-3-319-71246-8_11)
5. Durugkar, I., Gemp, I., Mahadevan, S.: Generative multi-adversarial networks. arXiv preprint [arXiv:1611.01673](https://arxiv.org/abs/1611.01673) (2016)
6. Hariharan, B., Arbeláez, P., Bourdev, L., Maji, S., Malik, J.: Semantic contours from inverse detectors. In: ICCV (2011)
7. Isola, P., Zhu, J.-Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: CVPR (2017)
8. Lin, G., Shen, C., Van Den Hengel, A., Reid, I.: Efficient piecewise training of deep structured models for semantic segmentation. In: CVPR (2016)
9. Liu, W., Rabinovich, A., Berg, A.C.: Parsenet: looking wider to see better. arXiv preprint [arXiv:1506.04579](https://arxiv.org/abs/1506.04579) (2015)
10. Liu, Z., Li, X., Luo, P., Loy, C.-C., Tang, X.: Semantic image segmentation via deep parsing network. In: ICCV, pp. 1377–1385 (2015)
11. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: CVPR (2015)
12. Luc, P., Couprie, C., Chintala, S., Verbeek, J.: Semantic segmentation using adversarial networks. In: NIPS Workshop on Adversarial Training (2016)

13. Luc, P., Neverova, N., Couprie, C., Verbeek, J., LeCun, Y.: Predicting deeper into the future of semantic segmentation. In: ICCV (2017)
14. Nguyen, T., Le, T., Vu, H., Phung, D.: Dual discriminator generative adversarial nets. In: NIPS (2017)
15. Shuai, B., Liu, T., Wang, G.: Improving fully convolution network for semantic segmentation. arXiv preprint [arXiv:1611.08986](https://arxiv.org/abs/1611.08986) (2016)
16. Souly, N., Spampinato, C., Shah, M.: Semi and weakly supervised semantic segmentation using generative adversarial network. arXiv preprint [arXiv:1703.09695](https://arxiv.org/abs/1703.09695) (2017)
17. Yu, F., Koltun, V.: Multi-scale context aggregation by dilated convolutions. arXiv preprint [arXiv:1511.07122](https://arxiv.org/abs/1511.07122) (2015)
18. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: CVPR (2017)



# Efficient Bayesian Optimisation Using Derivative Meta-model

Ang Yang<sup>(✉)</sup>, Cheng Li, Santu Rana, Sunil Gupta, and Svetha Venkatesh

Center for Pattern Recognition and Data Analytics,  
Deakin University, Geelong, Australia  
{leon.yang, cheng.l, santu.rana, sunil.gupta,  
svetha.venkatesh}@deakin.edu.com

**Abstract.** Bayesian optimisation is an efficient method for global optimisation of expensive black-box functions. However, the current Gaussian process based methods cater to functions with arbitrary smoothness, and do not explicitly model the fact that most of the real world optimisation problems are well-behaved functions with only a few peaks. In this paper, we incorporate such shape constraints through the use of a derivative meta-model. The derivative meta-model is built using a Gaussian process with a polynomial kernel and derivative samples from this meta-model are used as extra observations to the standard Bayesian optimisation procedure. We provide a Bayesian framework to infer the degree of the polynomial kernel. Experiments on both benchmark functions and hyperparameter tuning problems demonstrate the superiority of our approach over baselines.

**Keywords:** Bayesian optimisation · Gaussian process  
Meta learning · Derivative-based

## 1 Introduction

Bayesian optimization (BO) of black-box function [1] often use Gaussian Process (GP) as priors of latent functions. A GP is specified by a mean function and a covariance function. The squared exponential (SE) kernel is a popular choice of covariance function [2]. The posterior distribution is computed by combining the likelihood of these observations and GP prior. Then a utility function which combines the mean and variance of posterior GP is used to determine the next point for evaluating the black-box function.

Most real world functions, however resulting either from physical experiments or hyperparameter tuning, are well behaved. They are smooth and have a small number of local peaks. If such knowledge can be harnessed, then BO may converge faster. BO algorithms for well-behaved functions have been addressed only in limited contexts when either the function is monotonic [3] or it has a concave/convex shape [4]. BO methods for functions with more general shape properties such as incorporating the knowledge that the function has only a few peaks has not been addressed before, and thus remains an open problem.

Addressing that, we propose a new method that can flexibly incorporate the shape of the function through a derivative meta-model. The derivative meta-model is built using a polynomial. To maintain the Bayesian flavor and to maintain the ability of estimating the meta-model from a few observations, we use a Gaussian process with polynomial kernel (GPPK) for the meta-model. Based on the observed data we fit the GPPK and then sample derivative values for the use in the main GP for the BO. In effect, the main GP is built based on a trade-off between the flexible model induced by the stationery kernel and the structure induced by the derivative information based on GPPK. We refrain from using the samples of the function values from GPPK because we only want to pass the shape information through derivative, while keeping the function values guided mostly by the main GP. The crucial in this scheme is setting the degree of the polynomial kernel. We use a Bayesian formulation to estimate the degree from the observed data. We then use a truncated geometric prior, cut-off at degree of 10 and then normalised, which essentially prefers lower degree as a prior information. Posterior is then computed based on the marginal likelihood of the GPPK on the observed data. The mode of the posterior is then used as the degree for our derivative meta-model.

We demonstrate our method on three synthetic examples and applications on hyperparameter tuning for two machine learning algorithms. We compare with BO without derivatives and BO with true derivatives in synthetic examples and only compare BO without derivatives in hyperparameter tuning since true derivatives are not available in this case. In all experiments our proposed method outperforms the baselines. In summary, our contributions are: 1. Proposal of a new method to incorporate shape information in BO through a derivative meta-model; 2. Derivation of a mechanism to estimate the parameter of the prior shape function through Bayesian inference; 3. Validation on synthetic functions and applications of hyperparameter tuning.

## 2 Related Background

### 2.1 Bayesian Optimisation

Bayesian optimisation has two main components. The first is to model the unknown function using GP as a prior. The other component is to search the next point where to perform the experiment. The search for the next point is guided by a surrogate utility function, called acquisition function.

**Gaussian Process.** We briefly review GP [2] here. GP is a strategy of specifying prior distributions over the space of smooth functions. It is a distribution over function and the properties of the Gaussian distribution allow us to compute the predictive mean and variance in the closed form. GP is specified by its mean function  $\mu(\mathbf{x})$  and covariance function  $k(\mathbf{x}, \mathbf{x}')$ . A sample from a GP is a function given as:  $f(\mathbf{x}) \sim \mathcal{N}(\mu(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$  where  $\mathcal{N}$  is a Gaussian distribution and  $\mathbf{x}$  denotes a  $D$ -dimensional covariate vector. Without any loss in generality, the

prior mean function can be assumed to be a zero function making the GP fully defined by the covariance function. A popular choice of kernel is the squared exponential function given as:  $k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp(-\frac{1}{2} \frac{\|\mathbf{x} - \mathbf{x}'\|^2}{\rho_l^2})$  where  $\rho_l$  is the characteristic length scale, and  $\sigma_f$  is the signal standard deviation.

We denote a set of observations  $\mathcal{D} = \{\mathbf{x}_{1:t}, \mathbf{f}_{1:t}\}$ , where  $\mathbf{f}_{1:t} = \{f(\mathbf{x}_i)\}_{i=1}^t$ . The joint distribution of observations  $\mathcal{D}$  and a new observation  $\{\mathbf{x}_{t+1}, f_{t+1}\}$  is still a Gaussian. If the observation is a noisy estimate of the actual function value then  $y = f(\mathbf{x}) + \xi$  where  $\xi \sim N(0, \sigma_{noise}^2)$ . Then the predictive distribution of  $f_{t+1}$  can be written as  $\mathcal{P}(f_{t+1} | \mathcal{D}_{1:t}, \mathbf{x}_{t+1}) = \mathcal{N}(\mu(\mathbf{x}_{t+1}), \sigma^2(\mathbf{x}_{t+1}))$  with mean and variance:  $\mu(\mathbf{x}_{t+1}) = \mathbf{k}^T [K + \sigma_{noise}^2 I]^{-1} \mathbf{f}_{1:t}$ ,  $\sigma^2(\mathbf{x}_{t+1}) = k(\mathbf{x}_{t+1}, \mathbf{x}_{t+1}) - \mathbf{k}^T [K + \sigma_{noise}^2 I]^{-1} \mathbf{k}$  where  $\mathbf{k} = [k(\mathbf{x}_{t+1}, \mathbf{x}_1) \ k(\mathbf{x}_{t+1}, \mathbf{x}_2) \ \dots \ k(\mathbf{x}_{t+1}, \mathbf{x}_t)]$  and  $K$  is the kernel matrix given by:

$$K = \begin{bmatrix} k(\mathbf{x}_1, \mathbf{x}_1) & \dots & k(\mathbf{x}_1, \mathbf{x}_t) \\ \vdots & \ddots & \vdots \\ k(\mathbf{x}_t, \mathbf{x}_1) & \dots & k(\mathbf{x}_t, \mathbf{x}_t) \end{bmatrix} \tag{1}$$

**Acquisition Functions.** Acquisition functions have been defined so that it effectively trades off exploitation and exploration. Exploitation means the areas where the mean prediction for function values are high. Exploration means the areas where the epistemic uncertainty about the function values are high. In this paper, we use EI as the criteria. Assume that our optimisation problem is maximising  $f(\mathbf{x})$  and the current maximum is  $f(\mathbf{x}^+)$ . The improvement function  $I(\mathbf{x})$  [5] is written as:  $I(\mathbf{x}) = \max\{0, f(\mathbf{x}) - f(\mathbf{x}^+)\}$ . The analytic form of  $E(I(\mathbf{x}))$  can be obtained as [5]:

$$E(I(\mathbf{x})) = \begin{cases} (\mu(\mathbf{x}) - f(\mathbf{x}^+))\Phi(z) + \sigma(\mathbf{x})\phi(z) & \text{if } \sigma(\mathbf{x}) > 0 \\ 0 & \text{if } \sigma(\mathbf{x}) = 0 \end{cases}$$

where  $z = (\mu(\mathbf{x}) - f(\mathbf{x}^+)) / \sigma(\mathbf{x})$ .  $\Phi(z)$  and  $\phi(z)$  are the CDF and PDF of standard normal distribution.

### 3 Framework

We propose a new method to incorporate shape information about the objective function through the use of a derivative meta-model. First, we describe the construction of the meta-model using Gaussian process with a polynomial kernel (GPPK). Next, we construct our method of sampling the derivative information from the GPPK and then use it in the main Gaussian process. Finally, we present a Bayesian approach to estimate the polynomial degree based on the observations.

### 3.1 Meta-model

Some early researches have investigated the use of polynomial curve fitting in optimisation. Specifically, in [6] the authors theoretically explained the mechanism of curve fitting in global optimisation of expensive black box functions. Motivated from the usefulness of prior shape information [7], in our framework, we use Gaussian process with polynomial kernel to fit the observation data and then estimate derivative information.

**Gaussian Process with Polynomial Kernel (GPPK).** Gaussian processes allow us to compute the predictive mean and variance in closed form which is fully defined by the covariance function, The kernel in GPPK is defined as  $k(\mathbf{x}, \mathbf{x}') = (c + \mathbf{x} \cdot \mathbf{x}')^d$ , where  $c$  is kernel offset and  $d$  is the degree of the polynomial. The covariance matrix  $K$  can be computed by following Eq. (1) and the mean function of GPPK can be computed as:

$$\mu(\mathbf{x}) = \mathbf{k}K^{-1}\mathbf{y} \tag{2}$$

where  $\mathbf{k} = [(c + \mathbf{x} \cdot \mathbf{x}_1)^d (c + \mathbf{x} \cdot \mathbf{x}_2)^d \dots (c + \mathbf{x} \cdot \mathbf{x}_t)^d]$ .

**Derivative Estimation.** Now we can estimate the derivative values at our observations by differentiating the mean function Eq. (2)

$$\nabla \mathbf{f} = \frac{\partial}{\partial \mathbf{x}} \mu(\mathbf{x}) = \mathbf{k}'K^{-1}\mathbf{y} \tag{3}$$

where  $\mathbf{k}' = [d(c + \mathbf{x} \cdot \mathbf{x}_1)^{d-1} \cdot \mathbf{x}_1 \ d(c + \mathbf{x} \cdot \mathbf{x}_2)^{d-1} \cdot \mathbf{x}_2 \dots \ d(c + \mathbf{x} \cdot \mathbf{x}_t)^{d-1} \cdot \mathbf{x}_t]$ .

### 3.2 BO with Estimated Derivatives

Since the derivatives of a Gaussian Process is still a GP [8], the joint distribution of function values and derivatives is analytically tractable. In terms of squared exponential covariance function, the covariance between function values and partial derivatives can be written as [3]:

$$cov(f^i, \frac{\partial f^j}{\partial x_g^{(j)}}) = \sigma_f^2 exp(-\frac{1}{2} \sum_{b=1}^D \rho_l^{-2} (x_b^{(i)} - x_b^{(j)})^2) \times (\rho_l^{-2} (x_g^{(i)} - x_g^{(j)}))$$

and covariance between partial derivatives is given as:

$$cov(\frac{\partial f^j}{\partial x_g^{(i)}}, \frac{\partial f^j}{\partial x_h^{(j)}}) = \sigma_f^2 exp(-\frac{1}{2} \sum_{b=1}^D \rho_l^{-2} (x_b^{(i)} - x_b^{(j)})^2) \times \rho_l^{-2} (\delta_{gh} - \rho_l^{-2} (x_h^{(i)} - x_h^{(j)})(x_g^{(i)} - x_g^{(j)}))$$

where  $\delta_{gh} = 1$  if  $g = h$ , and  $\delta_{gh} = 0$  if  $g \neq h$ .

---

**Algorithm 1.** Bayesian Optimisation using Derivative Meta-model (BODMM)
 

---

- 1: **for**  $n = 1, 2, \dots, t$  **do**
  - 2: Fit the data  $\mathcal{D}$  using GPPK
  - 3: Estimate derivative values  $\nabla \mathbf{f}$  from GPPK via Eq.(3)
  - 4: Build GP with function observations and estimated derivatives of observations
  - 5: Find  $\mathbf{x}_{t+1}$  by maximising  $\mathbf{x}_{t+1} = \operatorname{argmax}_x EI(\mathbf{x}|\mathcal{D})$
  - 6: Evaluate the objective function:  $y_{t+1} = f(\mathbf{x}_{t+1}) + \xi$
  - 7: Augment the observation set  $\mathcal{D} = \mathcal{D} \cup (\mathbf{x}_{t+1}, y_{t+1})$ .
  - 8: **end for**
- 

Now using GP we can derive the posterior over a new function value  $f_{t+1}$  at  $\mathbf{x}_{t+1}$  when given a set of observations of the function values and a set of derivative information. We use  $\bar{K}_{[f_{1:t}, \nabla f_{1:t}]}$  to denote the joint covariance matrix over a set of observations of function values and the estimated derivatives. Then the new joint distribution for  $[f_{1:t}, \nabla f_{1:t}, f_{t+1}]$  is:

$$\begin{bmatrix} \mathbf{f}_{1:t} \\ \nabla \mathbf{f}_{1:t} \\ f_{t+1} \end{bmatrix} \sim \mathcal{N} \left( 0, \begin{bmatrix} \bar{K}_{[f_{1:t}, \nabla f_{1:t}]} & \bar{\mathbf{k}} \\ \bar{\mathbf{k}}^T & k(\mathbf{x}_{t+1}, \mathbf{x}_{t+1}) \end{bmatrix} \right) \quad (4)$$

where  $\bar{\mathbf{k}} = [k_{[f_{1:t}, \nabla f]}^T, k_{[f_{t+1}, \nabla f]}]^T$ , and the predictive distribution on  $\mathbf{x}_{t+1}$  is a normal distribution  $\mathcal{N}(\bar{\mu}, \bar{\sigma}^2)$  where  $\bar{\mu}(\mathbf{x}_{t+1})$  and  $\bar{\sigma}^2(\mathbf{x}_{t+1})$  are given as:

$$\bar{\mu}(\mathbf{x}_{t+1}) = \bar{\mathbf{k}}^T \bar{K}_{[f_{1:t}, \nabla f_{1:t}]}^{-1} [f_{1:t}, \nabla f_{1:t}] \quad (5)$$

$$\bar{\sigma}^2(\mathbf{x}_{t+1}) = k(\mathbf{x}_{t+1}, \mathbf{x}_{t+1}) - \bar{\mathbf{k}}^T \bar{K}_{[f_{1:t}, \nabla f_{1:t}]}^{-1} \bar{\mathbf{k}} \quad (6)$$

We then use the Eqs. (5) and (6) to construct acquisition function and perform BO. The proposed method is described in Algorithm 1.

### 3.3 Degree Estimation

Given the degree  $d = 2$  in the polynomial kernel, the posterior mean function in GP can be maximum quadratic. While a quadratic meta-model can be sufficient in majority of cases, we provide a mechanism to estimate the degree of the polynomial if one wishes so. To achieve this, in this subsection we infer the degree through Bayesian inference.

In Bayesian inference, the posterior probability of a variable is proportional to the product of the prior and the likelihood. In our case, the posterior of the degree  $d$  is mathematically computed as

$$p(d | X, \mathbf{y}) \propto p(d) p(\mathbf{y} | X, d) \quad (7)$$

The prior  $p(d)$  represents our belief on the degree. Since the degree is discrete, we choose the geometric distribution as our prior. The geometric distribution



presents the probability that the first occurrence of success requires  $m$  independent trials, each with success probability  $q$ ,

$$p(d = m) = (1 - q)^{m-1} q \quad (8)$$

where  $m = 1, 2, 3 \dots$ . In practice, we do not expect that the  $d$  is over than a high value such as 10 and then we can use the truncated geometric distribution with normalisation. The likelihood  $p(\mathbf{y} | X, d)$  in Eq. (7) is the marginal likelihood of GP with polynomial kernel. We compute it as following

$$\log p(\mathbf{y} | X, d) = -\frac{1}{2} \mathbf{y}^T (K + \sigma^2 I)^{-1} \mathbf{y} - \frac{1}{2} \log |K + \sigma^2 I| - \frac{n}{2} \log 2\pi \quad (9)$$

Given the Eqs. (8) and (9), we can compute the posterior as in Eq. (7), and then use the mode of the posterior as the estimated degree. Once we infer the  $d$ , we directly apply it into the derivative estimation as in Eq. (2).

## 4 Experiments

We firstly examine the capability of the GPPK to catch function shape. The results show that GPPK can approximately capture the 1D and 2D functions. We then evaluate our method on three different benchmark functions and then real world applications on hyperparameter tuning for two machine learning algorithms. We compare the proposed method BODMM with the following baselines for benchmark functions:

- Bayesian Optimisation without derivative observations (Standard BO).
- Bayesian Optimisation using true derivative values (BOTD).

As the true derivative values are not available in real applications, we only compare with Standard BO.

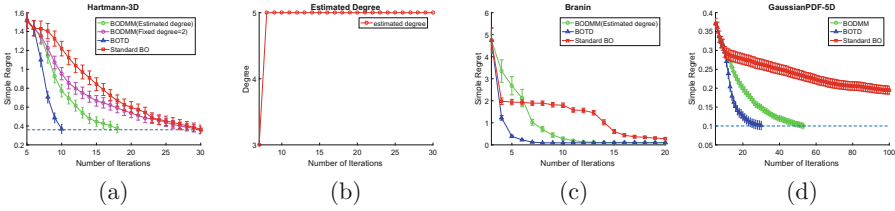
In all experiments, we use EI as the acquisition function and the SE kernel as the covariance function. We use DIRECT [9] to optimise the acquisition function. In terms of kernel parameters, we use the isotropic length scale  $\sigma_l = 0.1$ , signal variance  $\sigma_f^2 = 1$  and noise variance  $\sigma_{noise}^2 = (0.01)^2$ . In GPPK, we use  $c = 0.5$  as kernel offset and  $q = 0.5$  in truncated geometric distribution. We run each algorithm 100 trials with different initialisation and report the simple regret and standard errors for benchmark functions while reporting accuracy for hyperparameter tuning tasks. Simple regret is defined as  $r_t = f(\mathbf{x}^*) - f(\mathbf{x}_t^+)$  where  $f(\mathbf{x}^*)$  is the global optimum and  $f(\mathbf{x}_t^+) = \max_{\mathbf{x} \in \{\mathbf{x}_{1:t}\}} f(\mathbf{x})$  which is the current best value.

### 4.1 Experiment with Benchmark Test Functions

We test our algorithm on three benchmark functions as below:

1. 3D Hartmann (Hartmann-3D). The global minimum is  $f(\mathbf{x}^*) = -3.86278$  at  $\mathbf{x}^* = (0.114614, 0.555649, 0.852547)$  where search space is in  $[0, 1]$ ;

2. 2D Branin’s function (Branin-2D). The global minimum is  $f(\mathbf{x}^*) = 0.397887$  at  $\mathbf{x}^* = (\pi, 2.275)$  where search space is in  $[0, 4]$ .
3. Unnormalized 5D Gaussian PDF (Gaussian PDF-5D). The global maximum is  $f(\mathbf{x}^*) = 1$  at  $\mathbf{x}^* = (1, 1, 1, 1, 1)$  where search space is in  $[0, 2]$ .



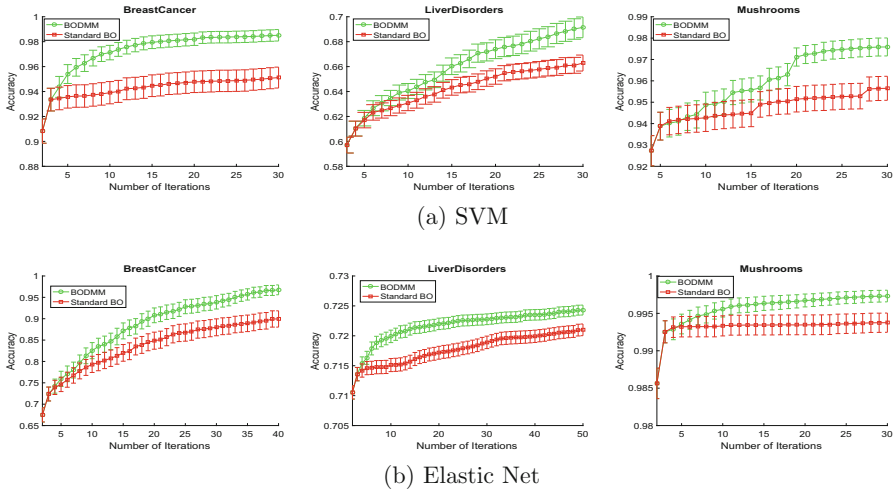
**Fig. 1.** Simple regret vs iterations for (a) Hartmann-3D function (b) Estimated degree  $d$  in the GPPK for optimising Hartmann-3D function (c) Branin’s Function (d) GaussianPDF-5D.

We set a convergence threshold which is reaching 10% to the optimum. We did not apply the threshold on Branin’s since the BO for it can converge fast. We start to examine our algorithm on Hartmann-3D. Figure 1a plots the simple regret vs iteration for three different algorithms. For the proposed BODMM we run it with fixed degree and estimated degree respectively. BO using true derivative values performs the best in all three algorithms and converges after 10 iterations. It is easy to understand since true derivatives have been incorporated in this algorithm. Our algorithm with fixed degree and estimated degree outperforms Standard BO. The setting with estimated degree performs better than that of fixed degree. Figure 1b demonstrates the estimated degree at each iteration. We also receive positive results from other test functions. Results of Branin’s function and Gaussian PDF-5D have been illustrated in Fig. 1c and d respectively.

### 4.2 Hyperparamter Tuning

We experiment with three real world datasets for tuning hyperparameters of two classifiers: Support Vector Machines (SVM) and Elastic Net. In SVM we optimise two hyperparameters which are the cost parameter ( $C$ ) and the width of the RBF kernel ( $\gamma$ ). The search bounds for the two hyperparameters are  $C = 10^\lambda$  where  $\lambda \in [-3, 3]$  and  $\gamma = 10^\omega$  where  $\omega \in [-5, 0]$  correspondingly. To make our search bounds manageable, we optimise for  $\lambda$  and  $\omega$ . In Elastic Net, the hyperparameters are the  $l_1$  and  $l_2$  penalty weights. The search bound for both of them is  $[10^{-5}, 10^{-2}]$ . We optimise in the range of exponents ( $[-5, -2]$ ). All three datasets: BreastCancer, LiverDisorders and Mushrooms are publicly available from UCI data repository [10].

The results for hyperparameter tuning are showing in Fig. 2. In all cases our approach BODMM performs better than Standard BO. For example in the



**Fig. 2.** Accuracy vs iterations for (a) hyperparameter tuning for SVM on three datasets: BreastCancer, LiverDisorders and Mushrooms (b) hyperparameter tuning for Elastic Net on the same three datasets.

leftmost graphic of Fig. 2b, Standard BO achieves to **0.89** after 40 iterations while our algorithm achieves to **0.97**.

## 5 Conclusion

We propose a novel method for Bayesian optimisation for well-behaved functions with small numbers of peaks. We incorporate this information through a derivative meta-model. The derivative meta-model is based on a Gaussian process with polynomial kernel. By controlling the degree of the polynomial we control the shape of the main Gaussian process which is built using the SE kernel and the covariance matrix is computed by using both the observed function value and the derivative values sampled from the meta-model. We also provide a Bayesian way to estimate the degree of the polynomial based on a truncated geometric prior. In experiments, both on benchmark test functions and the hyperparameter tuning from popular machine learning models, our proposed model converged faster than the baselines.

**Acknowledgment.** This research was partially funded by the Australian Government through the Australian Research Council (ARC) and the Telstra-Deakin Centre of Excellence in Big Data and Machine Learning. Professor Venkatesh is the recipient of an ARC Australian Laureate Fellowship (FL170100006).

## References

1. Jones, D.R., Schonlau, M., Welch, W.J.: Efficient global optimization of expensive black-box functions. *J. Global Optim.* **13**(4), 455–492 (1998)
2. Rasmussen, C.E., Williams, C.K.: *Gaussian Processes for Machine Learning*, vol. 1. MIT Press, Cambridge (2006)
3. Riihimäki, J., Vehtari, A.: Gaussian processes with monotonicity information. In: *Proceedings of the Thirteenth International Conference on AIStat.* (2010)
4. Jauch, M., Peña, V.: Bayesian optimization with shape constraints. arXiv preprint [arXiv:1612.08915](https://arxiv.org/abs/1612.08915) (2016)
5. Mockus, J.: Application of bayesian approach to numerical methods of global and stochastic optimization. *J. Global Optim.* **4**(4), 347–365 (1994)
6. Denison, D.G.T., Mallick, B.K., Smith, A.F.M.: Automatic Bayesian curve fitting. *J. R. Stat. Soc. Ser. B (Stat. Methodol.)* **60**(2), 333–350 (1998)
7. Abdolmaleki, A., Lioutikov, R., Peters, J.R., Lau, N., Reis, L.P., Neumann, G.: Model-based relative entropy stochastic search. In: *NIPS* (2015)
8. Solak, E., Murray-Smith, R., Leithead, W.E., Leith, D.J., Rasmussen, C.E.: Derivative observations in gaussian process models of dynamic systems. In: *NIPS* (2003)
9. Finkel, D.E.: *Direct optimization algorithm user guide*. CRSC (2003)
10. Dheeru, D., Karra Taniskidou, E.: *UCI Machine Learning Repository* (2017)



# Prior Knowledge Guided Gene-Disease Associations Prediction: An Enhanced Inductive Matrix Completion Approach

Lei Chen<sup>1,2,3</sup>(✉), Jianyu Pu<sup>1</sup>, Ziwen Yang<sup>1</sup>, and Xingguo Chen<sup>1</sup>

<sup>1</sup> School of Computer Science,  
Nanjing University of Posts and Telecommunications, Nanjing, China  
chenlei@njupt.edu.cn

<sup>2</sup> Jiangsu High Technology Research Key Laboratory for Wireless  
Sensor Networks, Nanjing, China

<sup>3</sup> College of Computer Science and Technology,  
Nanjing University of Aeronautics and Astronautics, Nanjing, China

**Abstract.** Exploring gene-disease associations is of great significance for early prevention, diagnosis and treatment of diseases. Most existing methods depend on specific type of biological evidence and thus are limited in the application. More importantly, these methods ignore some inherent prior sparsity and structure knowledge which is useful for predicting gene-disease associations. To address these challenges, a novel Enhanced Inductive Matrix Completion (EIMC) model is proposed to predict pathogenic genes by introducing the prior sparsity and structure knowledge into the traditional Inductive Matrix Completion (IMC). Specifically, the EIMC model not only employs the sparse regularization to preserve the prior sparsity of gene-disease associations, but also employs the manifold regularization to capture the prior structure information of data distribution. To the best of our knowledge, the proposed EIMC is the first model to simultaneously incorporate both prior sparse and manifold regularizations into the same objective function. Additionally, note that our proposed EIMC model also integrates the features of genes and diseases extracted from various types of biological data, and can predict new genes and diseases by using an inductive learning strategy. Finally, the extensive experimental results demonstrate that our proposed model outperforms other state-of-the-art methods.

**Keywords:** Gene-disease associations prediction · Matrix Completion  
Prior sparse regularization · Prior manifold regularization · Inductive learning

## 1 Introduction

Deciphering the gene-disease associations is important for the diseases' early prevention, clinical diagnosis and efficient treatment. Moreover, in the biomedicine field, discovering the pathogenic gene as soon as possible plays an important role in the research and development of effective therapeutic drugs, and its economic benefits are also enormous. The early studies of gene-disease association was based on clinical

biopsy, which usually cost a lot of labor and material resources, and may bring the risk of infection to the patient. This situation not only greatly limits the development of pathogenic gene research, but also seriously affects the quality of public datasets. Currently, very few reliable gene-disease associations are reported in the public databases such as the Online Mendelian Inheritance in Man (OMIM) [1] and the Genetic Association Database (GAD) [2].

Recently, with the rapid development of high-throughput sequencing technique, various useful biological information, such as gene array information, gene intrinsic characteristics, and similarity information among genes or diseases, can be easily obtained. The emergence of such information provides an opportunity to study new gene-disease associations predicting methods. Zhao et al. [3] developed the Katz based on network similarity measurement method, which constructs a gene-disease heterogeneous network by incorporating gene-gene network, disease-disease network and gene-disease associations. Wu et al. [4] proposed the CIPHER method, which assumes that phenotypically similar diseases are caused by functionally related genes and computes a score to assess the likelihood that a gene relate to the specific disease. Li and Patra [5] applied the random walk on the heterogeneous network which connect gene network and disease network by gene-disease associations. All these similarity measurement based methods can integrate different biological networks to increase the amount of information, but cannot predict the nodes outside the networks, and the predictive effectiveness depends on the construction of high-quality networks. Thus, some researchers start to leverage the machine learning techniques to overcome these limitations. For instance, Singh-Blom et al. [6] proposed the CATAPULT method by introducing the biased SVM classifiers into the traditional Katz approach to improve the prediction accuracy. Natarajan and Dhillon [7] proposed an inductive gene-diseases prediction method by incorporating the features of gene and disease from various biological information into the IMC (Inductive Matrix Completion) model developed by Jian and Dhillon [8], which can overcome the cold-start problem faced in the standard matrix completion (MC) model.

However, all these existing approaches ignore the prior knowledge of biological data, such as the prior sparsity of the gene-disease associations and the prior structure information of data distribution. Specifically, there are only a few specific pathogenic genes for each disease in real-world, which leads to the inherent sparsity of the gene-disease associations. The prior structure information of data distribution manifests as gene correlation consistency and disease correlation consistency. Gene correlation consistency means that genes cause the disease or similar diseases are neighbor to each other in the gene network, and their similarity is close. Disease correlation consistency means that similar diseases have the same or similar pathogenic genes, and the distance between them is small in the disease network. To address these challenges, based on the IMC method, we propose an Enhanced Inductive Matrix Completion (EIMC) method, which exploits the prior sparse regularization to preserve the prior sparsity of the gene-disease associations and uses the prior manifold regularization to capture the prior structure information of data distribution.

## 2 Method

### 2.1 Inductive Matrix Completion

Our goal is to discover the pathogenic gene by predicting gene-disease associations. Here, constructing the gene-disease matrix  $M \in \mathbb{R}^{m \times n}$ , where each row represents a gene, and each column represents a disease, that is,  $M_{ij} = 1$  if gene  $i$  is associated to disease  $j$ , and  $M_{ij} = 0$  otherwise. Natarajan et al. [7] adopt the IMC model to predict the gene-disease associations. This model assumes that gene-disease associations matrix is generated by multiplying the feature matrices of genes and diseases to a low-rank matrix  $X$ . The goal of the method is to recover  $X$  by exploiting the observations from  $M$ . Specifically, let  $A \in \mathbb{R}^{f_g \times m}$  be the feature matrix of genes, where each column denotes the  $f_g$  features of a gene. Let  $B \in \mathbb{R}^{f_d \times n}$  be the feature matrix of diseases, where each column denotes the  $f_d$  features of a disease. The IMC model aims to recover a low-rank matrix  $X \in \mathbb{R}^{f_g \times f_d}$  using observations from the gene-disease matrix  $M$ . In this paper, we let  $M = A^T X B$ , and the goal of this method is to learn  $X$  utilizing  $\Omega$  (i.e. the set of observed entries). This IMC model can be formulated as follows:

$$\min_{X \in \mathbb{R}^{f_g \times f_d}} \|X\|_* \quad s.t. \quad P_\Omega(A^T X B) = P_\Omega(M) \quad (1)$$

where  $\|X\|_*$  denotes the nuclear norm of  $X$ . Obviously, the IMC model can predict new genes as well as new diseases by integrating their features, so it is inductive.

### 2.2 Enhanced Inductive Matrix Completion with Both Prior Sparse and Manifold Regularizations

As a matter of fact, there are only a few specific pathogenic genes for each disease in real-world, which leads to the inherent sparsity of the gene-disease associations. The IMC model ignores this prior sparsity which can be used to improve the prediction. Correspondingly, we introduce the  $L_1$ -norm of matrix  $A^T X B$  into the IMC model to fit the inherent sparsity of the gene-disease associations. Thus the improved IMC model can be formulated as followings:

$$\min_{X \in \mathbb{R}^{f_g \times f_d}} \|X\|_* + \mu \|A^T X B\|_1 \quad s.t. \quad P_\Omega(A^T X B) = P_\Omega(M) \quad (2)$$

where  $\|A^T X B\|_1$  denotes the  $L_1$ -norm of  $A^T X B$ , and  $\mu > 0$  is the parameter used to adjust the sparsity degree of the gene-disease associations matrix.

Furthermore, by analyzing the biological data, it is easy to observe that some genes and diseases have correlation consistency, i.e., the genes that cause the same or similar diseases are neighbor to each other in the gene network, and their similarity is close. Also, some diseases with the same or similar pathogenic genes have shorter distance between them in the disease network. This correlation consistency describes the mutual dependence among genes and diseases. However, the aforementioned models ignore such prior manifold information of data distribution, which is useful for predicting gene-disease associations. To address this issue, we use the gene-gene similarity matrix

to model the gene correlation consistency, and the disease-disease similarity matrix to model the disease correlation consistency. Here we employ the popular manifold regularization technique [9] to model the two correlation consistencies.

Specifically, for a given gene-gene similarity matrix  $S_g \in \mathbb{R}^{m \times m}$ , we can construct an adjacent graph  $G_g$  with  $m$  nodes, where each node  $v_i$  denotes to the gene  $g_i$ . When  $g_i$  is among  $k$  nearest neighbors of  $g_j$  or  $g_j$  is among  $k$  nearest neighbors of  $g_i$ , an edge is added between nodes  $v_i$  and  $v_j$ . With the  $G_g$ , we can define a weight matrix  $W_g$  to model the gene correlation consistency, where  $W_g(i, j) = S_g(i, j)$  if  $v_i$  connects to  $v_j$ , and  $W_g(i, j) = 0$  otherwise. Thus, the prior gene correlation consistency can be modeled as the following manifold regularization term:

$$\begin{aligned} & \min_{X \in \mathbb{R}^{f_g \times f_d}} \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m W_g(i, j) \|A^T X B(i, :) - A^T X B(j, :)\|^2 \\ & = \min_{X \in \mathbb{R}^{f_g \times f_d}} \text{Tr}(B^T X^T A (D_g - W_g) A^T X B) = \min_{X \in \mathbb{R}^{f_g \times f_d}} \text{Tr}(B^T X^T A L_g A^T X B) \end{aligned} \tag{3}$$

where  $D_g$  is the diagonal matrix and  $D_g(i, i) = \sum_j W_g(i, j)$ ,  $L_g = D_g - W_g$  is the Laplacian matrix of genes, and  $\text{Tr}(\cdot)$  denotes the trace norm of a matrix. Similarly, for a given disease-disease similarity matrix  $S_d \in \mathbb{R}^{n \times n}$ , we can model the prior disease correlation consistency by using the following manifold regularization term:

$$\min_{X \in \mathbb{R}^{f_g \times f_d}} \text{Tr}(A^T X B L_d B^T X^T A) \tag{4}$$

where  $L_d = D_d - W_d$  is the Laplacian matrix of diseases,  $W_d$  is a weight matrix to model the disease correlation consistency and  $W_d(i, j) = S_d(i, j)$ ,  $D_d$  is the diagonal matrix and  $D_d(i, i) = \sum_j W_d(i, j)$ .

Therefore, based on the aforementioned analysis, by incorporating both the prior sparse and manifold regularizations into the IMC model, we can obtain an Enhanced Inductive Matrix Completion (EIMC) model as follows:

$$\begin{aligned} & \min_{X \in \mathbb{R}^{f_g \times f_d}} \|X\|_* + \mu \|A^T X B\|_1 + \lambda/2 (\text{Tr}(B^T X^T A L_g A^T X B) + \text{Tr}(A^T X B L_d B^T X^T A)) \\ & \text{s.t. } P_\Omega(A^T X B) = P_\Omega(M) \end{aligned} \tag{5}$$

where  $\mu$  and  $\lambda$  are two tunable positive constants to control the contribution of prior sparse regularization as well as prior manifold regularization, their values can be determined by cross-validation on the training data. Furthermore, For ease of description, we introduce an intermediate variable  $C$  to replace  $A^T X B$ , then, the EIMC model can be reformulated as the following equivalent optimizing problem:

$$\begin{aligned} & \min_{X, C} \|X\|_* + \mu \|C\|_1 + \frac{\lambda}{2} (\text{Tr}(C^T L_g C) + \text{Tr}(C L_d C^T)) \\ & \text{s.t. } P_\Omega(C) = P_\Omega(M) \quad C = A^T X B \end{aligned} \tag{6}$$



### 2.3 Optimizing EIMC Using ADMM

In this paper, we employ the Alternating Direction Method of Multipliers (ADMM) [10] to design an optimization algorithm for solving EIMC in Eq. (6). The key steps of this algorithm are to iteratively optimize the following four Sub-problems:

$$\begin{cases} X^{k+1} = \operatorname{argmin}_X L(X, C^k, Y_1^k, Y_2^k) & (7a) \\ C^{k+1} = \operatorname{argmin}_C L(X^{k+1}, C, Y_1^k, Y_2^k) & (7b) \\ Y_1^{k+1} = Y_1^k + \rho_1 P_\Omega(C^{k+1} - M) & (7c) \\ Y_2^{k+1} = Y_2^k + \rho_2 (C^{k+1} - A^T X^{k+1} B) & (7d) \end{cases} \quad (7)$$

where  $L(X, C^k, Y_1^k, Y_2^k)$  is the augmented Lagrange function corresponds to Eq. (6),  $Y_1$  and  $Y_2$  are Lagrange multipliers,  $\rho_1 > 0$  and  $\rho_2 > 0$  are penalized parameters.

We solve Sub-problem 1 in Eq. (7a) and Sub-problem 2 in Eq. (7b) by employing the Proximal Forward Backward Splitting (PFBS) method [11], with its convergence being proven by Combettes et al. [11]. Specifically, for each iteration  $t$ , the following iterative sequence can be used to solve Sub-problem 1:

$$X_{t+1}^k = \mathbf{D}_{\delta_X}(X_t^k - \delta_X \nabla F(X_t^k)) \quad (8)$$

where  $\delta_X = 2/\rho_2 \sigma_{\max}(AA^T) \sigma_{\max}(BB^T)$  denotes the updating step size,  $\sigma_{\max}(\cdot)$  is the maximum singular value of matrix,  $\mathbf{D}_{\delta_X}(\cdot)$  denotes the proximal operator of the nuclear norm [12], and  $\nabla F(X_t^k)$  is the gradient of  $F(X)$  at  $X_t^k$ :

$$F(X) = \rho_2/2 \|C^k - A^T X B + Y_2^k/\rho_2\|_F^2 \quad (9)$$

Similarly, the iterative sequence for solving Sub-problem 2 are as following:

$$C_{t+1}^k = \mathbf{S}_{\mu\delta_C}(C_t^k - \delta_C \nabla G(C_t^k)) \quad (10)$$

where  $\delta_C = (\rho_1^2 + \rho_2^2 + \lambda^2 \sigma_{\max}^2(L_g) + \lambda^2 \sigma_{\max}^2(L_d))^{-1/2}$  denotes the updating step size,  $\mathbf{S}_{\mu\delta_C}$  denotes the proximal operator of the  $L_1$ -norm [12], and  $\nabla G(C_t^k)$  is the gradient of  $G(C)$  at  $C_t^k$ :

$$G(C) = \left\{ \begin{aligned} & \lambda/2 (Tr(C^T L_g C) + Tr(C L_d C^T)) + \rho_1/2 \|P_\Omega(C - M) + Y_1^k/\rho_1\|_F^2 \\ & + \rho_2/2 \|C - A^T X^{k+1} B + Y_2^k/\rho_2\|_F^2 \end{aligned} \right\} \quad (11)$$

Theoretically, for the jointly convex problem with the two optimization variables, He and Yuan [13] has demonstrated that the ADMM method is guaranteed to converge as long as all Sub-problems are solvable. In our EIMC model, obviously, the objective function in Eq. (7) is jointly convex for  $X$  and  $C$ . Based on this fact, our proposed optimization algorithm also has the provable convergence.

### 3 Results

#### 3.1 Dataset

The dataset used in the experiments can be download from <http://marcottelab.org/index.php/Catapult>, which is collected from OMIM to predict gene-disease associations. This dataset involves 12331 genes and 3209 diseases, which includes 3954 known gene-disease associations, one pair-wise gene-gene similarity matrix, and one pair-wise disease-disease similarity matrix. Usually, we can employ the dimensionality reduction methods to process various high-dimension biological data, and thus obtain the corresponding low-dimension compact features. In this paper, the PCA (Principal Component Analysis) technique is used to extract low-dimension compact features from high-dimension gene microarray data and clinical phenotype data. In addition, we also extract the leading eigenvectors from the similarity matrices as latent features of genes and diseases. Finally, we concatenate the two feature vectors from aforementioned different data sources into one vector. In this way, each gene and disease can be represented in terms of one 300-dimension and 200-dimension feature vector respectively.

#### 3.2 Evaluation Method

To validate our proposed EIMC method, we performed extensive experiments by also comparing with the three different competing methods, including Katz [6], MC [7] and IMC [7]. All the involved parameters in these methods are optimized by using the 3-fold cross-validation procedure. Here we determine the candidate pathogenic genes by adopting top- $r$  ranking strategy which was widely used in associations analysis field. We measure the prediction performance of gene-disease associations in terms of Recall and Precision at different threshold  $r$ . It is desirable to obtain a good prediction effect by using a small threshold in the biological research field, so we taking  $r \leq 100$  to report the experimental results. In addition, considering that some ‘popular’ genes and diseases have more chances to be predicted so that leads to the inflated recall, we do not only evaluate the overall prediction performance on all genes and diseases, but also exploit the alone prediction performance on new genes and new diseases. Specifically, we use the genes or diseases with only one known association in the dataset to validate the performance of discovering new genes or new diseases.

#### 3.3 Overall Performance

In this sub-section, we assess the overall performance of gene-disease association prediction. The optimal parameters are determined by using grid searching strategy. Figure 1 reports the experimental results of the 3 competing methods and our proposed method, where the vertical axis in Fig. 1(a) denotes the recall, and the horizontal axis denotes different thresholds  $r$ . From Fig. 1(a), we can see that our EIMC method consistently outperforms other state-of-the-art methods for all  $r$  values. Especially, for the top-100 prediction, our method achieves 25.9% in terms of recall metric, which is better than those of the competing methods (MC with 6.6%, Katz with 11.3%, and IMC

with 23.2%). Figure 1(b) shows the precision-recall curves of all four methods. From Fig. 1(b), we can see that when recall is over 4%, the precision of our method is consistently better than other methods. All these experimental results demonstrate that the prior sparsity and structure knowledge is useful to improve the predicting performance of gene-disease associations.

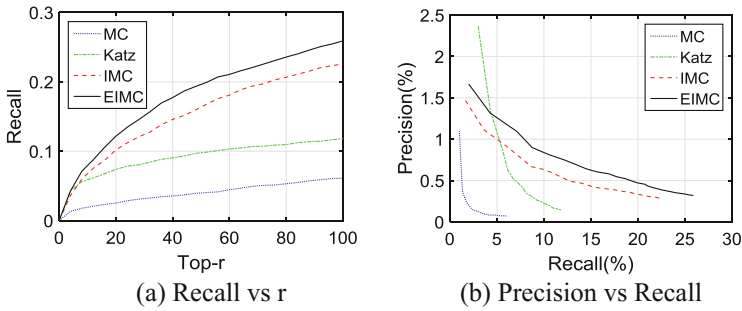


Fig. 1. Overall performance at different threshold  $r$

### 3.4 New Genes and New Diseases

As we all know, the ‘popular’ genes and diseases would impact on the prediction performance, which means that genes or diseases well connected have more chances to be predicted, and thus lead to the inflated recall. To address this problem, we conduct a set of experiments to evaluate the discovery performance for the new genes and new diseases (i.e. genes or diseases with only one association in the dataset). Figure 2 shows the recall performance at different threshold  $r$  for new genes and new diseases. From Fig. 2(a), we can see that Katz is better than EIMC in terms of recall for new genes when  $r \leq 25$ , because in a small threshold range, the genetic information extracted from the heterogeneous network is more conducive to predicting. However, when  $r > 25$ , our EIMC method achieves the better recall performance since it combines the prior sparsity and structure distribution information. Furthermore, from Fig. 2(b), we can see that the

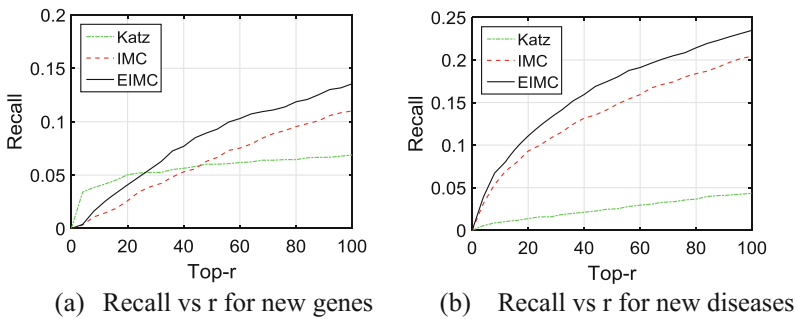


Fig. 2. Recall performance at different threshold  $r$  for new genes and new diseases

proposed EIMC method also achieves the better recall for new diseases than other methods. Here we do not use MC as the competing method since MC can't predict new genes or new diseases.

## 4 Conclusion

Deciphering the gene-disease associations is important to understand disease mechanisms. In this paper, we proposed a novel prior knowledge guided Enhanced Inductive Matrix Completion (EIMC) method. Our method do not only utilize the prior sparse regularization to preserve the prior sparsity of gene-disease associations, but also exploits the prior manifold regularization to capture the prior structure information of data distribution. The extensive experiments show that our proposed method consistently outperforms other state-of-the-art competing methods. In addition, our method can also predict new genes and new diseases in an inductive learning way. Thus, the proposed EIMC can be an effective tool for biologists to explore the new gene-disease associations. Future work will focus on integrating more biological data to further improve the prediction performance of gene-disease associations.

**Acknowledgements.** This work is supported by the National Natural Science Foundation of China (grant number 61572263), the Natural Science Foundation of Jiangsu Province (grant number BK20161516), the Postdoctoral Science Foundation of China (grant number 2015M581794), the Postdoctoral Science Foundation of Jiangsu Province (grant number 1501023C).

## References

1. Hamosh, A., Scott, A.F., Amberger, J.S., Bocchini, C.A., McKusick, V.A.: Online mendelian inheritance in Man (OMIM), a knowledgebase of human genes and genetic disorders. *Nucleic Acids Res.* **33**(1), D514–D517 (2005)
2. Becker, K.G., Barnes, K.C., Bright, T.J., Wang, S.A.: The genetic association database. *Nat. Genet.* **36**(5), 431–432 (2004)
3. Zhao, J., Yang, T.H., et al.: Ranking candidate disease genes from gene expression and protein interaction: a Katz-centrality based approach. *PLoS ONE* **6**(9), e24306 (2011)
4. Wu, X., Jiang, R., Zhang, M.Q., Li, S.: Network-based global inference of human disease genes. *Mol. Syst. Biol.* **4**(1), 189 (2008)
5. Li, Y., Patra, J.C.: Genome-wide inferring gene-phenotype relationship by walking on the heterogeneous network. *Bioinformatics* **26**(9), 1219–1224 (2010)
6. Singh-Blom, U.M., Natarajan, N., Tewari, A., Woods, J.D., Dhillon, I.S., Marcotte, E.M.: Prediction and validation of gene-disease associations using methods inspired by social network analyses. *PLoS ONE* **8**(5), e5897 (2013)
7. Natarajan, N., Dhillon, I.S.: Inductive matrix completion for predicting gene–disease associations. *Bioinformatics* **30**(12), 60–68 (2014)
8. Jain, P., Dhillon, I.S.: Provable inductive matrix completion. *arXiv:1306.0626* (2013)
9. Chen, L., Yang, G., et al.: Correlation consistency constrained matrix completion for web service tag refinement. *Neural Comput. Appl.* **26**(1), 101–110 (2015)

10. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J.: Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.* **3**(1), 1–122 (2010)
11. Combettes, P.L., Wajs, V.R.: Signal recovery by proximal forward-backward splitting. *SIAM J. Multiscale Model. & Simul.* **4**(4), 1168–1200 (2005)
12. Cai, J.F., Candès, E.J., Shen, Z.: A singular value thresholding algorithm for matrix completion. *SIAM J. Optim.* **20**(4), 1956–1982 (2010)
13. He, B.S., Yuan, X.M.: On the  $O(1/n)$  convergence rate of Douglas-Rachford alternating direction method. *SIAM J. Numer. Anal.* **50**(2), 700–709 (2012)



# Text Classification with Enriched Word Features

Jingda Xu, Cheng Zhang, Peng Zhang, and Dawei Song<sup>(✉)</sup>

School of Computer Science and Technology, Tianjin University, Tianjin, China  
{jingdaxu, zccode, pzhang, dwsong}@tju.edu.cn

**Abstract.** Text classification is a fundamental task in natural language processing. Most existing text classification models focus on constructing sophisticated high-level text features but ignore the importance of word features. Those models only use low-level word features obtained from a linear layer as input. To explore how the quality of word representations affects text classification, we propose a deep architecture which can extract high-level word features to perform text classification. Specifically, we use different temporal convolution filters, which vary in size, to capture different contextual features. Then a transition layer is used to coalesce the contextual features and form an enriched high-level word representations. We also find that word feature reuse is useful in our architecture to enrich word representations. Extensive experiments on six publically available datasets show that enriched word representations can significantly improve the performance of classification models.

**Keywords:** Text classification · Enriched word representation  
Temporal convolution

## 1 Introduction

Text classification aims at labeling natural language texts with relevant categories from a predefined set. It is widely used in information filtering, sentiment analysis and many other relevant applications [1].

In text classification, feature representation is an important and fundamental problem. It can be divided into two parts, namely word representation and document (or sentence) representation. While word representation aims to represent each word with a vector that would contain certain semantic and syntactic information [10], document representation aims to use a vector to capture the semantics of the whole text [9].

Recently, various text classification approaches based on neural networks have been proposed. An example is the Recursive Neural Network which has been applied in sentiment classification [12]. It captures the semantics of a sentence via a tree structure. Moreover, Recurrent Neural Network (RNN) and its variants, such as LSTM and GRU, are widely applied in text classification [15, 17]. Convolutional Neural Network (CNN) has also been used for text

classification. A representative is [7] which adopts a rather shallow CNN. The model uses only one convolution layer followed by a max pooling layer over time.

Some deeper CNN architectures were also proposed for text classification. Kalchbrenner proposed a model which introduces k-max pooling and uses two convolution layers to capture a hierarchical feature of the input text [6]. The model proposed by Zhang used six temporal convolution layers, followed by three fully connected classification layers [16]. Conneau also used a deeper CNN with 49 temporal convolution layers to perform text classification [2].

However, nearly all models mentioned above focus on using sophisticated architecture to extract high-level text features to improve the performance of classification model. While the word representation used in models is only extracted from a very shallow architecture, one linear layer. Such word representation can only be seen as a low-level feature which may not contain enough semantics. There have few works use deep architecture to capture high-level sophisticated word features to perform text classification.

As words are the basic unit of text, a powerful word representation would lead to a more effective representation of document. We would like to stress that word feature representation plays a key role in text classification models. In this paper, we propose a deep architecture consisted of stacked multi-channel temporal convolution layers and a transition layer to construct an enriched word representation in an end-to-end model and named it as *MCTC*. We also find word feature reuse, which is a common operation in computer vision field (such as ResNet [3] and DenseNet [4]), is also effective in enriching word representations.

We carry out extensive experiments on six publically available datasets including short text and long text classification. To highlight the role of word representations in text classification, we only use a max pooling operation to capture text feature and feed it into a fully connected layer, followed by a softmax function that predicts the probability distribution over classes. The experiments show that enriched word representations can have richer semantics and perform better than traditional word embedding. Moreover, without using any complex text representation methods, our model also shows superior performances over most of baseline methods.

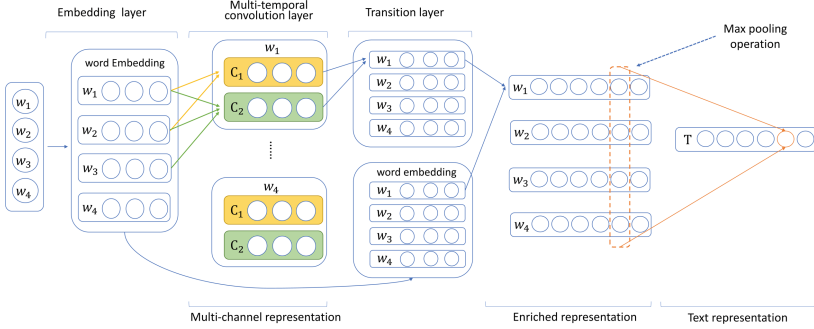
## 2 Model Description

In this section, we present the detail of learning an enriched word representation. An overview of the approach is displayed in Fig. 1.

### 2.1 Enriched Word Representations

We first describe the detail of learning enriched word representation. The input text  $W$  is formalized as a sequence of  $n$  words:

$$W = [w_1, w_2, \dots, w_n] \quad (1)$$



**Fig. 1.** The architecture of our *MCTC* architecture applied on a four words sentence. We set the dimension of word embeddings with size  $E = 3$ . The first layer is a normal embedding layer that converts each one-hot word representation into a real-valued vector. For the sake of illustration, the architecture of model only contains one temporal convolution layer. The temporal convolution layer has two filters with varying window size. Transition layer is a  $1 \times 1$  convolution layer which is used to reduce the number of feature-maps.

where  $w_i \in \mathbb{R}^V$  is an one-hot representation, and  $V$  is the size of vocabulary. Then, each word is projected into a low dimensional vector space by a randomly initialized word look up table  $D \in \mathbb{R}^{E \times V}$ , and  $E$  is the dimensionality of a word vector. We denote the corresponding word embedding for  $i$ -th word as:

$$e_i = Dw_i \tag{2}$$

Next, we apply a temporal convolution with multiple filters on each word and its context to capture contextual features.

Let  $f_{i:i+j}$  refer to the concatenation of word features, i.e.,  $[f_i, f_{i+1}, \dots, f_{i+j}]$ . Let  $h_m$  be the window size of  $m$ -th filters, where  $m \in \{1, 2, \dots, M\}$ . We define  $f_{(i-\lfloor \frac{h_m-1}{2} \rfloor):i}$  as the left context and  $f_{i:(i+\lceil \frac{h_m-1}{2} \rceil)}$  as the right context of  $f_i$ . Since the meaning of a word is typically depend on its context, we aim to obtain richer semantic information for each word to enrich its representation. To do so, we propose to use different convolution filters to capture contextual features from different scale context and regard those features as different channel of a word representation. For example, the multi-channel vector representation of the  $i$ -th word in the input text is calculated as follows:

$$f_{i_m} = A(\text{Conv}(W_m, e_{(i-\lfloor \frac{h_m-1}{2} \rfloor):(i+\lceil \frac{h_m-1}{2} \rceil)})) \tag{3}$$

where  $A(\cdot)$  is a non-linear function, and  $\text{Conv}(\cdot)$  represents the multi-channel temporal convolution operation. The temporal convolution filters that are parameterized by tensor  $W_m \in \mathbb{R}^{c \times k_o \times h_m \times k_i}$ . We define  $c$  as the channel number of the input feature-maps,  $k_i$  as the dimensionality of input feature map, and  $k_o$  as the dimensionality of the output feature map. In our experiments, we set  $k_o = k_i = E$  for all convolution filters to ensure each channel of word representation has the same size.



However, not all the contextual features are useful and effective to enrich the word representation. Therefore, we use a  $1 \times 1$  convolution filter to automatically learn which feature is helpful. The convolution filter is parameterized by tensor  $W_t \in \mathbb{R}^{c_{in} \times c_{out} \times 1 \times 1}$ , where  $c_{in}$  and  $c_{out}$  represent the numbers of input and output feature-maps respectively. To reduce the channel of enriched word representation to 1, we set  $c_{out} = 1$ . Following the notation of a  $1 \times 1$  convolution layer given in [4], we also named it as transition layer. The word representation of the transition layer is computed as follows:

$$f_i^t = Conv(W_t, f_i) \quad (4)$$

Given the text embedding matrix, the text vector representation is computed by applying a column-wise max-pooling and then fed into a linear classifier to predict the probability of each label. Since the max-pooling is a parameter-free operation, we can think that the classification accuracy is only affected by the quality of the word representation.

## 2.2 Word Feature Reuse

Feature reuse is an effective operation in CV field by adding additional connection between previous layers and subsequent layers. Feature reuse can also alleviate the vanishing-gradient problem and strengthen feature propagation in a deep neural network. However, there have been no work demonstrates that feature reuse is also effective in NLP tasks. To explore whether feature reuse is useful in our model, we densely connect each convolution layer as described in DenesNets [4]. For each layer, the feature-maps of all preceding layers are used as inputs, and the current layer’s own feature-maps are used as inputs into all subsequent layers.

To further explore the ability of feature reuse, we use the original word embedding as the other feature of the word. Thus, each word in the text has two vector representations. One representation is the word embedding from the input-layer and the other is obtained from *MCTC* architecture. We use the highway network [14] to coalesce those two word features to form a final word representation.

$$x = f_i \oplus f_i^t \quad (5)$$

$$f_i^c = H(x, W_H) \cdot T(x, W_T) + x \cdot (1 - T(x, W_T)) \quad (6)$$

where  $\oplus$  represents the concatenate operation.  $H(\cdot)$  and  $T(\cdot)$  are non-linear functions (parameterized by  $W_H$  and  $W_T$ ). We omit the biases for simplicity and clarity.

## 3 Datasets and Experimental Setup

We evaluate our model on six freely available datasets: MR, SST, TREC, AG’s news, Yelp Review Polarity and Yelp Review Full. Table 1 provides detailed information about each dataset.

**Table 1.** A summary of the datasets, including the number of classes, the number of train/test set entries, the vocabulary size, the average text length and the task type. (CV means there was no standard train/test split and thus 10-fold CV was used).

Datasets	Classes	Train samples	Test samples	Vocabulary size	Average length	Task type
MR	2	11.9k	CV	18.8k	20	Sentiment analysis
SST	2	10.7k	2.2k	17.8k	18	Sentiment analysis
TREC	6	5.9k	0.5k	9.5k	10	Question classification
AG’s news	4	120k	7.6k	70k	35	News categorization
Yelp Review Full	5	650k	50k	238k	122	Sentiment analysis
Yelp Review Polarity	2	560k	38k	240k	121	Sentiment analysis

- **Movie Review (MR).** The dataset contains movie reviews (with one sentence per review) introduced by [11]. The classification task involves detecting positive/negative reviews.
- **Stanford Sentiment Treebank (SST).** The dataset contains movie reviews parsed and labeled by [12]. The classification task involves detecting positive/negative reviews. The data is provided at the phrase level. In experiments, both phrases and sentences are used in training, but only sentences are scored at test time.
- **Trec Question Classification (Trec).** The task involves classifying a question into 6 types such as entity, human, location, numeric value, etc.
- **AG’s news (AG).** The dataset is a collection of more than 1 million news articles. We use the dataset pre-processed by [16] which have 4 classes.
- **Yelp Review Full (Yelp5).** The Yelp reviews dataset is obtained from the Yelp Dataset Challenge in 2015. This dataset predicts the number of stars that the user has given (from 0 to 4).
- **Yelp Review Polarity (Yelp2).** The dataset is same as Yelp Review Full but predicts a polarity label (positive/negative).

### 3.1 Implementation Details

On all datasets, we set the dimension of word embedding to 128. Before entering the temporal convolution layer, each feature-map is zero-padded to keep the size of output the same as input’s. For long text classification, we use three filters of width 3/5/7, and for short text classification we use filters of width 2/3. For the regularization we employ drop-out [13] on the embedding layer and transition layer with the same probability  $p = 0.5$ . The network is trained with mini-batches by backpropagation and the gradient-based optimization is performed by using the Adam update rule [8].

### 3.2 Baseline Models

We compared our method with several state-of-the-art approaches. The metric for evaluating each model is the accuracy of prediction.

- **CNN-Rand**. This method [7] uses a one-layer CNN with randomly initialized word embeddings for text classification.
- **Dynamic CNN**. This model [6] uses five convolution layers and K-max pooling operation with pre-trained word embeddings.
- **Char CNN**. It uses a 12-layer convolutional neural network with only character-level features as input [16].
- **VDCNN**. This method use a deep temporal CNN architecture (29 layers) to perform text classification. The input of this model is also on character-level [2].
- **fastText**. This method averages the embeddings of all words in text to form a text representation [5].
- **WE**. Similar to fastText, but use max-pooling instead of average-pooling.

## 4 Results and Discussion

We compare the performance of enriched word representations and traditional word representations with max-pooling operation in text classification. Moreover, we also compare our method with several state-of-the-art approaches which aim to extract sophisticated text features. The results are listed in Table 2.

**Table 2.** Results of our MCTC method against other methods.

Short text	MR	SST	Trec	Long Text	AG	Yelp5	Yelp2
CNN-Rand [7]	76.1	82.7	91.2	VDCNN [2]	91.3	64.7	95.7
Dynamic-CNN [6]	-	86.8	93.0	Char CNN [16]	87.2	62.0	94.7
WE	74.3	82.5	89.2	WE	91.1	58.3	93.1
fastText	75.0	83.0	89.7	fastText	91.5	60.4	93.9
MCTC	77.3	85.4	90.4	MCTC	91.8	64.4	96.1
MCTC-dense	<b>80.5</b>	<b>86.1</b>	<b>93.0</b>	MCTC-dense	<b>92.2</b>	<b>65.7</b>	<b>96.2</b>

We can see from the table that, when compared to the models which only use traditional word embeddings and pooling operation to perform text classification, the enriched word representations bring a great improvement in both short text classification and long text classification. We think there are some reasons for this result. Since pooling operation is parameter-free, the accuracy is only influenced by the quality of word representation. The texts usually contain some polysemes and typos which is difficult to model word meaning for traditional word embedding method. Moreover, it's difficult for traditional word embedding method to model the words which are unfrequent in the corpus. Since the words with similar context tends to have similar meaning, our method utilizes the contexts to enrich the meaning of the words and can effectively overcome those difficulties.

When compare the other methods which use deep architectures with traditional word embeddings to extract sophisticated text features, our method also perform competitively and even get a better result in some datasets. The experiments show that high-quality word representations are as important as sophisticated text features in text classification.

#### 4.1 Learned Word Representations

We explore the word representations learned by our models on the Yelp Review Polarity dataset. Table 3 has the nearest neighbors of word representations learned from the *MCTC* layer. We compare the representations obtained from the traditional embedding layer and representations after transition layer.

**Table 3.** Nearest neighbor words (based on cosine similarity) of word representations.

Method	Unpleasant	Unsatisfied	Helpful	Unwilling	Nice	Suitable
WE	Rude	Argued	Awesome	Disappointment	Good	Comfortable
	Acted	Overrated	Great	Mediocre	Friendly	Cool
	Disgusting	Tasteless	Fantastic	Pitiful	Great	Flavorful
	Positives	Unacceptable	Cosy	Insulting	Fresh	Fresh
MCTC	Rude	Disappointing	Tasty	Refused	Fun	Greeting
	Irritating	Disappointment	Friendly	Uninspiring	Cool	Impress
	Horrific	Unattentive	Smooth	Overpricing	Beautiful	Cleaned
	Lacking	Disappointed	Efficient	Thinnest	Friendly	Special

The *MCTC* architecture extract contextual features to enrich the word representation, so the same word may have different representation when its context is changed. To obtain a stable representation for each word, we average the representations of word with different contexts. We can see from the table that enriched word representation is better in capturing sentiment information. For example, in the traditional word embedding space, the nearest neighbors of *unpleasant* are *rude*, *acted*, *disgusting*, *positives*, which contain some neutral or positive words. However, in the enriched word representation space, the nearest neighbors of *unpleasant* is all negative words.

## 5 Conclusions

In this paper, we have proposed a deep architecture (*MCTC*) to enrich word representations for text classification. By using multi-temporal convolution filters, we can get a multi-channel word representation which is able to incorporate richer contextual information. An enriched word representation can be obtained by coalescing the contextual features. We also explore the effect of word feature reuse in our model. We have compared our model with a range of strong baselines, on both short text and long text classification datasets. Our model achieves a remarkable performance which is better or comparable to the state-of-the-art.

## References

1. Aggarwal, C.C., Zhai, C.: A survey of text classification algorithms. In: Aggarwal, C., Zhai, C. (eds.) *Mining Text Data*, pp. 163–222. Springer, Boston (2012). [https://doi.org/10.1007/978-1-4614-3223-4\\_6](https://doi.org/10.1007/978-1-4614-3223-4_6)
2. Conneau, A., Schwenk, H., Barrault, L., Lecun, Y.: Very deep convolutional networks for text classification. In: *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, vol. 1, pp. 1107–1116 (2017)
3. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778 (2016)
4. Huang, G., Liu, Z., Weinberger, K.Q., van der Maaten, L.: Densely connected convolutional networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2017)
5. Joulin, A., Grave, E., Bojanowski, P., Mikolov, T.: Bag of tricks for efficient text classification (2016). arXiv preprint [arXiv:1607.01759](https://arxiv.org/abs/1607.01759)
6. Kalchbrenner, N., Grefenstette, E., Blunsom, P.: A convolutional neural network for modelling sentences. In: *Proceedings of ACL*, pp. 655–665 (2014)
7. Kim, Y.: Convolutional neural networks for sentence classification (2014). arXiv preprint [arXiv:1408.5882](https://arxiv.org/abs/1408.5882)
8. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization (2014). arXiv preprint [arXiv:1412.6980](https://arxiv.org/abs/1412.6980)
9. Le, Q., Mikolov, T.: Distributed representations of sentences and documents. In: *International Conference on Machine Learning*, pp. 1188–1196 (2014)
10. Mikolov, T., Yih, W.t., Zweig, G.: Linguistic regularities in continuous space word representations. In: *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 746–751 (2013)
11. Pang, B., Lee, L.: Seeing stars: exploiting class relationships for sentiment categorization with respect to rating scales. In: *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pp. 115–124. Association for Computational Linguistics (2005)
12. Socher, R., et al.: Recursive deep models for semantic compositionality over a sentiment treebank. In: *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 1631–1642 (2013)
13. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **15**, 1929–1958 (2014). <https://doi.org/10.1214/12-AOS1000>
14. Srivastava, R.K., Greff, K., Schmidhuber, J.: Highway networks (2015). arXiv preprint [arXiv:1505.00387](https://arxiv.org/abs/1505.00387)
15. Tang, D., Qin, B., Liu, T.: Document modeling with gated recurrent neural network for sentiment classification. In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 1422–1432 (2015)
16. Zhang, X., Zhao, J., LeCun, Y.: Character-level convolutional networks for text classification. In: *Advances in Neural Information Processing Systems*, pp. 649–657 (2015)
17. Zhou, C., Sun, C., Liu, Z., Lau, F.: A C-LSTM neural network for text classification (2015). arXiv preprint [arXiv:1511.08630](https://arxiv.org/abs/1511.08630)



# Attention-Based Linguistically Constraints Network for Aspect-Level Sentiment

Jinyu Lu and Yuexian Hou<sup>(✉)</sup>

School of Computer Science and Technology, Tianjin University, Tianjin, China  
{jylu,yxhou}@tju.edu.cn

**Abstract.** Aspect-level sentiment analysis is an essential subtask of sentiment classification. It aims at classifying the sentiment polarity of given aspect in its context. Recently, a variety of deep learning models have been proposed to solve this task, such as Long Short-Term Memory Networks (LSTM), Convolutional Neural Networks (CNN). In particular, great improvement has been achieved by using attention mechanism. At the same time, the adoption of linguistic resources to improve sentiment classification has also drawn researchers' attention, and achieved state-of-the-art performance on traditional sentiment classification. Hence in this paper, we explore to combine linguistically constraints with attention mechanism to achieve comparable performance on aspect-level sentiment analysis. Experimental results on SemEval 2014 Datasets showed that the proposed model achieves good performance and verifies the effectiveness of linguistic resources on this task. To our knowledge, there are no work combining the attention mechanism and linguistic resources on this task before. This work gives inspirations to further research.

**Keywords:** Attention mechanism  
Aspect-level sentiment classification · Linguistically constraints LSTM

## 1 Introduction

Sentiment classification aims at classifying texts to different classes, such as positive, negative or more fine-grained classes, such as very positive, neutral, etc. Aspect-level sentiment classification is an important subtask of sentiment analysis [1–3]. Different from general sentiment analysis, it depends on the context of the text, but also on the information of given aspect. For instance, for the text “The plot of the film is very good, the music is not bad, but the actor is so bad”, the sentiment polarity of aspect “plot” is very positive, the sentiment polarity of aspect “music” is neutral, while the sentiment polarity is quite negative for the aspect “actor”. Therefore, even in the same sentence, sentiment polarities might be completely opposite for different aspects.

In recent years, deep learning methods have achieved state-of-the-art performance on lots of NLP tasks, such as machine translation [4], document classification [5] and question answer [6]. More recently, attention mechanism has

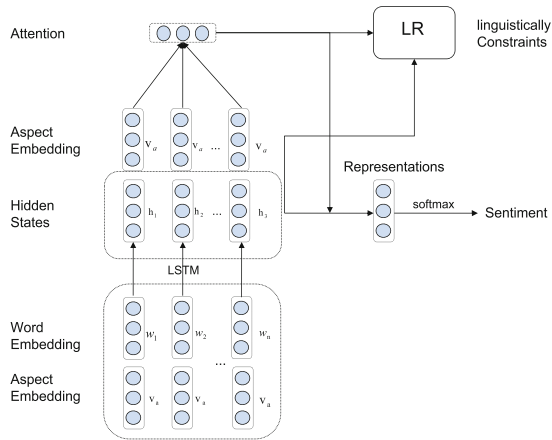
been widely used and obtained great improvement [1,2]. Besides, in [7], linguistic resources are adopted to solve traditional sentiment classification and improved the performance a lot. This work and other related works give us the inspiration to construct a model integrating attention mechanism and linguistically constraints into deep learning method.

The goal of this paper is fully employing linguistic resources to improve attention-based aspect-level sentiment classification. We use three types of linguistic resources in this paper: sentiment lexicon, negation lexicon and intensity lexicon. Because the sentiment lexicon tells us the polarity of a word, which can be used to determine the sentiment polarity, the word of negative lexicon can shift the polarity of sentiment classification, and the word of intensity lexicon can also influence the degree of sentiment polarity.

In summary, the main contribution of this research is introducing linguistic resources with attention mechanism to aspect-level sentiment classification. To the best of our knowledge, this is the first attention-based linguistically constraints model designed for aspect-level sentiment classification. Experimental results show that our model performs well for different datasets, and get comparable performance.

## 2 Attention-Based Linguistically Constraints LSTM Network (ALC-LN)

The overall architecture of ALC-LN is shown in Fig. 1. First, we adopt Glove vectors [8] as the initial setting of word embedding V. And we add the aspect embedding into the vector of each input word. In this way, the output hidden representations can get the information from the given aspect.



**Fig. 1.** The overall architecture of ALC-LN. The  $w_1, w_2, \dots, w_n$  are the word vector in a sentence whose length is  $n$ . The  $v_a$  is the aspect embedding. The  $h_1, h_2, \dots, h_n$  represent the hidden vector.

Next, we use the Long Short-Term Memory Network (LSTM) to learn the word representation. In short, in LSTM structure, the memory cell  $c_t$  and hidden state  $h_t$  is a function of previous hidden state  $h_{t-1}$ , previous memory cell  $c_{t-1}$  and the input  $x_t$ . It can be formulated as follows:

$$h_t, c_t = \sigma^{(LSTM)}(h_{t-1}, c_{t-1}, x_t), \tag{1}$$

the hidden state  $h_t$  indicate the representation of this position  $t$ . More details about LSTM can be found in [9].

Then, we use attention mechanism to get the weight value of each hidden state. The attention mechanism is designed to make the model know which part of the information in the input data is important in the training process. In this paper, we use the attention mechanism of [2], which achieved good performance in this task. Let  $H \in \mathbf{R}^{d \times n}$  be a matrix which is made up of the hidden states  $[h_1, ..h_n]$ , the  $d$  is the size of hidden layers and  $n$  is the length of input.  $v_a$  represents the aspect embedding and  $e_n \in \mathbf{R}^N$  is a column vector with  $N$ s.  $v_a \otimes e_N$  represents the operation that repeat the linearly transformed  $v_a$  as many times as the length of sentence. The vector of attention weight is  $\alpha$  and the weight of hidden representation is  $r$ . The conditional probability distribution is  $y$ .

$$M = \tanh\left(\begin{bmatrix} W_h H \\ W_v v_a \otimes e_N \end{bmatrix}\right), \tag{2}$$

$$\alpha = \text{softmax}(w^T M), \tag{3}$$

$$r = H\alpha^T, \tag{4}$$

$$h_{new} = \tanh(W_p r + W_x h_n), \tag{5}$$

$$y = \text{softmax}(W_s h_{new} + b_s), \tag{6}$$

where  $W_h, W_v, w^T, W_p, W_x, W_s, b_s$  are projection parameters.  $h_{new}$  is the final representation of a sentence that considers the given aspect.

### 2.1 Linguistically Constraints

At the same time, we combine linguistically constraints with attention mechanism to improve the performance. For aspect-level sentiment analysis, different parts of a sentence have different degrees of linguistically constraints weight. The weight value of linguistically constraint is the same as the weight value of hidden state at position  $t$ . For making linguistically constraints play a role in aspect-level sentiment classification, we defined a new loss function to train our model:

$$\mathcal{L}(\theta) = - \sum_i \hat{y}_i^j \log y_i^j + \alpha \sum_i \sum_t w_{t,i} C_{t,i} + \beta \sum ||\theta||^2, \tag{7}$$

where  $\hat{y}_i$  is the gold distribution for sentence  $i$ ,  $y_i$  is the predicted distribution, and  $j$  is the index of classes.  $C_{t,i}$  is one of the constraints or combination of different constraints on sentence  $i$ , the  $\alpha$  is the weight for the linguistically



constraints regularization term,  $\beta$  is the coefficient for  $L_2$  regularization,  $\theta$  is the parameter set. Then  $t$  is the word position of a sentence and  $w_{t,i}$  is the attention weight of linguistically constraints  $C_{t,i}$  at position  $t$ . The weight  $w_{t,i}$  is calculated by the attention mechanism.

**No-Sentiment Constraint (NSC):** NSC means that the sentiment distributions of adjacent positions should not be different much if the additional input word  $x_t$  is not a sentiment word. It is formulated as follows:

$$C_t^{(NSC)} = \max(0, D_{KL}(p_t || p_{t-1}) - m), \quad (8)$$

$m$  is the hyper parameter for margin,  $p_t$  is the predicted distribution at the state of position  $t$  and  $D_{KL}(p_t || p_{t-1})$  is the symmetric KL divergence which defined as follows:

$$D_{KL}(p || q) = \frac{1}{2} \sum_{l=1}^N (p(l) \log q(l) + q(l) \log p(l)), \quad (9)$$

where  $p, q$  are distributions on sentiment labels  $l$  and  $N$  is the number of labels.

**Sentiment Constraint (SC):** If the input word is found in sentiment lexicon, the sentiment distribution of the current position should be significantly different from the next or previous positions. We propose a polarity shifting distribution  $s_c \in R^C$  for each sentiment class which defined in the sentiment lexicon. It is formulated as follows:

$$p_{t-1}^{(SC)} = p_{t-1} + s_c(x_t), \quad (10)$$

$$C_t^{(SC)} = \max(0, D_{KL}(p_t || p_{t-1}^{(SC)}) - m). \quad (11)$$

The  $p_{t-1}^{(SC)}$  is the drifted sentiment distribution at position  $t$ ,  $c(x_t)$  is the prior sentiment class of word  $x_t$  and  $s_c$  is a parameter which will be optimized during training.

**Negation Constraint (NC):** NC defines how negation words transfer the sentiment distribution of the text. When the input word  $x_t$  is a negation word, the sentiment polarity will be transferred. We use a transformation matrix  $M_n \in R^{C \times C}$  for each negation word  $n$ . The hypothesis of this constraint is that the sentiment distribution of current position should be close to the next or the previous position with the transformation when the current position is a negation word. It is formulated as follows:

$$p_{t-1}^{(NC)} = \text{softmax}(M_{x_j} \times p_{t-1}), \quad (12)$$

$$p_{t+1}^{(NC)} = \text{softmax}(M_{x_j} \times p_{t+1}), \quad (13)$$

$$C_t^{(NC)} = \min \begin{cases} \max(0, D_{KL}(p_t || p_{t-1}^{(NC)}) - m) \\ \max(0, D_{KL}(p_t || p_{t+1}^{(NC)}) - m). \end{cases} \quad (14)$$

The  $p_{t-1}^{(NC)}$  and  $p_{t+1}^{(NC)}$  are the sentiment distributions after transformation, and the  $M_{x_j}$  is the transformation matrix for a negation word  $x_j$ , which is also a parameter that has to be learnt by the model.

**Intensity Constraint (IC):** Intensity words will influence the result of sentiment classification. When the input word  $x_t$  is a intensity word, the valence degree will be changed. The formulation of the intensity effect is quite the same as that in the negation constraint, but has different parameters. For the sake of brevity, We don't repeat the formula here.

### 3 Experiment

The dataset we used for evaluation is from SemEval 2014 [10], containing reviews of restaurant and laptop domains, which was widely used in previous works. We remove a few examples which have “conflict label”. So in our work, the set of class label now is  $\{positive, neural, negative\}$ . Our aim is to identify the aspect polarity of a sentence with the given aspect. The details of the dataset is present in Table 1. The linguistic resources we use is from [7].

**Table 1.** Statistic of the datasets

Dataset	Positive	Negative	Neutral
Laptop-train	994	870	464
Laptop-test	341	128	169
Restaurant-train	2164	807	637
Restaurant-test	728	196	196

#### 3.1 Comparison with Baseline Methods

For comprehensively evaluate the performance of ALC-LN. We compare our proposed framework with several baseline methods on both datasets.

**Table 2.** Accuracy on SemEval 2014 of different models

Model	Laptop	Restaurant
Majority	0.535	0.650
LSTM	0.665	0.744
TD-LSTM	0.681	0.756
AT-LSTM	0.689	0.762
AEAT-LSTM	0.687	0.772
ALC-LN	<b>0.704</b>	<b>0.787</b>

**Majority** is a basic baseline method, which assigns the majority sentiment polarity in training set to each instance in the test set. **LSTM** uses merely one LSTM network to model the context and get the hidden state of each word.

**TD-LSTM** uses two long short-term memory (LSTM) networks to model the left context with aspect and the right context with aspect respectively. **AT-LSTM** models the context words through LSTM networks, and combines aspect embedding with the word hidden states to supervise the generation of attention vectors. **AEAT-LSTM** is based on AT-LSTM. It appends the aspect embedding with each word embedding vector. We evaluate these models on the SemEval 2014 and the results are shown in Table 2.

It can be seen from Table 2 that the Majority method gets the worst result. Other methods are all neural network models and perform better than the Majority method, which indicates neural network has potentials to automatically generate representations and can effectively improve the performance of sentiment classification. LSTM method is the worst of all the neural network baseline methods, because it treats aspect and other context words equivalently and does not make full use of the given aspect information. The TD-LSTM outperforms LSTM about 1% and 2% on Laptop and Restaurant datasets respectively, since it derives from the standard LSTM and processes the left and right contexts with given aspect.

Further, AT-LSTM, AEAT-LSTM and ALC-LN stably exceed the TD-LSTM method because of the utilization of attention mechanism. They capture important parts of the context with the supervision of given aspect. AEAT-LSTM and ALC-LN get the better performance than AT-LSTM. The aspect embedding can account for the performance improvement. Appending the aspect embedding to each word vector can generate more reasonable representations for aspect-level sentiment classification.

It can be seen that ALC-LN achieves the best performance among all baseline methods. Compared with AEAT-LSTM model, ALC-LN improves the performance about 1.7% and 1.5% on the Laptop and Restaurant datasets respectively. The main reason is that the attention mechanism help the linguistically constraints play an active role in sentiment classification. It reveals that combine linguistic resources with attention mechanism can further improvement the attention-based model.

### 3.2 Analysis of ALC-LN Model

In order to investigate the effect of each individual constraint, we employ ablation experiments. We evaluated four variants of our final model which removes specifically one constraint from the full model. Besides, we also explore to reveal how attention mechanism influence the performance of linguistic constraints in aspect-level sentiment classification. We add a comparison model (ALC-LN(-AW)) which remove the attention weight  $w_{t,i}$  from the Eq. (6). The results were shown in Table 3, where we can see that the NSC and SC play a pivotal role, and NC and IC are effective but not important enough when compared with NSC and SC. And We can also see that accuracy decrease a lot without attention mechanism. The reason is that the high linguistically constraint value at position  $t$  are likely to be irrelevant even opposite to given aspect.

**Table 3.** The analysis for ALC-LN.

Model	Laptop	Restaurant
ALC-LN	<b>0.704</b>	<b>0.787</b>
ALC-LN(-NSC)	0.692	0.778
ALC-LN(-SC)	0.695	0.780
ALC-LN(-NC)	0.700	0.782
ALC-LN(-IC)	0.698	0.784
ALC-LN(-AW)	0.692	0.778

## 4 Related Work

Aspect-level sentiment classification is a fundamental task of sentiment analysis. The traditional way to solve this task is to manually design a set of features. With the abundance of sentiment lexicons [11], the lexicons-based features were built for sentiment analysis [12]. Most of these studies pay attention to build sentiment classifiers with features [13]. But the performance of these works are largely dependent on the quality of the features. In addition, the feature building works are labor-intensive.

Nowadays, neural networks [14] were used to solve sentiment analysis, and achieved state-of-the-art performance in aspect-level sentiment classification. Recently, attention mechanism methods have been used successfully in plenty of areas [15, 16]. However, these attention-based didn't utilize linguistic resources. Applying linguistic constraints to text classification can be seen in [17], which introduces three linguistically motivated structured regularizes for text categorization. Additionally the previous work [7] applied group linguistically regularizes to increase the performance of sentiment classification. Our work differs in that we combining the linguistically constraints with attention mechanism.

## 5 Conclusion

In this paper, we propose an attention-based linguistically constraints LSTM network (ALC-LN) for aspect-level sentiment classification. The main idea of ALC-LN is to apply linguist resources to aspect-level sentiment classification. The ALC-LN benefits from the combination of linguist resources and the attention mechanism. Results show that combining linguistically constraints with attention mechanism is able to improve the performance on aspect-level sentiment classification.

**Acknowledgments.** This work is funded in part by the National Key R&D Program of China (2017YFE0111900), the Key Project of Tianjin Natural Science Foundation (15JCZDJC31100), the National Natural Science Foundation of China (Key Program, U1636203), the National Natural Science Foundation of China (U1736103) and MSCA-ITN-ETN - European Training Networks Project (QUARTZ).

## References

1. Chen, P., Sun, Z., Bing, L., et al.: Recurrent attention network on memory for aspect sentiment analysis. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pp. 452–461 (2017)
2. Wang, Y., Huang, M., Zhao, L.: Attention-based LSTM for aspect-level sentiment classification. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pp. 606–615 (2016)
3. Ma, D., Li, S., Zhang, X., et al.: Interactive attention networks for aspect-level sentiment classification. arXiv preprint [arXiv:1709.00893](https://arxiv.org/abs/1709.00893) (2017)
4. Tan, Z., Su, J., Wang, B., et al.: Lattice-to-sequence attentional Neural Machine Translation models. *Neurocomputing* **284**, 138–147 (2018)
5. Yang, Z., Yang, D., Dyer, C., et al.: Hierarchical attention networks for document classification. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 1480–1489 (2016)
6. Yin, W., Yu, M., Xiang, B., et al.: Simple question answering by attentive convolutional neural network. arXiv preprint [arXiv:1606.03391](https://arxiv.org/abs/1606.03391) (2015)
7. Qian, Q., Huang, M., Lei, J., et al.: Linguistically regularized LSTMs for sentiment classification. arXiv preprint [arXiv:1611.03949](https://arxiv.org/abs/1611.03949) (2016)
8. Pennington, J., Socher, R., Manning, C.: Glove: global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543 (2014)
9. Hochreiter, S., Schmidhuber, J.: Long short-term memory. *Neural Comput.* **9**(8), 1735–1780 (1997)
10. Pontiki, M., Galanis, D., Pavlopoulos, J., et al.: SemEval-2014 Task 4: aspect based sentiment analysis. In: Proceedings of International Workshop on Semantic Evaluation, pp. 27–35 (2014)
11. Perez-Rosas, V., Banea, C., Mihalcea, R.: Learning sentiment lexicons in Spanish. In: LREC 2012, p. 73 (2012)
12. Mohammad, S.M., Kiritchenko, S., Zhu, X.: NRC-Canada: building the state-of-the-art in sentiment analysis of tweets. arXiv preprint [arXiv:1308.6242](https://arxiv.org/abs/1308.6242) (2013)
13. Mullen, T., Collier, N.: Sentiment analysis using support vector machines with diverse information sources. In: Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (2004)
14. Mikolov, T., Sutskever, I., Chen, K., et al.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems, pp. 3111–3119 (2013)
15. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. arXiv preprint [arXiv:1409.0473](https://arxiv.org/abs/1409.0473) (2014)
16. Rush, A.M., Chopra, S., Weston, J.: A neural attention model for abstractive sentence summarization. arXiv preprint [arXiv:1509.00685](https://arxiv.org/abs/1509.00685) (2015)
17. Yogatama, D., Smith, N.A.: Linguistic structured sparsity in text categorization. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), vol. 1, pp. 786–796 (2014)



# Personalized POIs Travel Route Recommendation System Based on Tourism Big Data

Chenzhong Bin<sup>1</sup>, Tianlong Gu<sup>2</sup>, Yanpeng Sun<sup>2</sup>, Liang Chang<sup>2(✉)</sup>,  
Wenping Sun<sup>2</sup>, and Lei Sun<sup>2</sup>

<sup>1</sup> School of Information and Communication,  
Guilin University of Electronic Technology, Guilin 541004, China  
binchenzhong@163.com

<sup>2</sup> Guangxi Key Lab of Trusted Software,  
Guilin University of Electronic Technology, Guilin 541004, China  
{cctlgu, changl}@guet.edu.cn,  
{yanpeng\_sun, sunlei92}@yeah.net, 173852394@qq.com

**Abstract.** One of the most important travel preparation activities for tourists is to plan a personalization POIs travel route to a new city, and it is yet a challenging task however. In this paper, we first propose a novel method to integrating multi-source tourism big data on websites to generate POIs knowledgebase and POIs visit sequences. Then a POI-Visit pattern sequence mining algorithm is proposed to generate various candidate POIs travel route. Last, the POIs travel route recommendation method is designed to provide a list of personalization POIs travel routes under tourist personal constraints, including the travel duration, the type of companion in trip, the visit season and the preferring tourism types etc. To validate the proposed system, extensive experiments are conducted on real tourism data set related to the city of Guilin in China, which contains 10,109 real travelogues, 132 POIs profiles and 8,646 POI traffic times.

**Keywords:** POI travel route recommendation · Sequential pattern mining  
Travelogue · Tourism big data

## 1 Introduction

Nowadays, travel recommendation systems play an important role in providing convenient tourism information to tourists. Although existing works are efficient in travel recommendations [1], there are two problems in providing personalized POIs itineraries for various tourists. First, the existing researches usually use single type of tourism information to generate travel recommendations, which are lack of personality and rationality. Second, previous systems provide tourism recommendations without considering various tourism attributes, such as preference, travel duration, visit season and who is accompany in trips etc.

In this work, we propose a personalized POIs travel route recommendation system to solve the above problems. Our contributions are summarized as follows. (1) The multi-source tourism information integrating method is designed to generate structured POIs visit sequences. (2) The POI-Visit pattern sequence mining algorithm is proposed to discover diverse frequent POIs travel routes from the generated POIs visit sequences. (3) A POIs travel route recommendation procedure is proposed to retrieve and rank a list of candidate itineraries under the personal constraints of a tourist, meanwhile ensure the route have a considerable travel value.

## 2 Related Works

In tourism recommending scenario, the cold start or the data sparsity problems are more serious than traditional recommending scenarios. Therefore, in recent researches, mining collective intelligence from user-generated content to enhance the personalization of travel recommendations is a promising approach to alleviate this situation.

In [2], researchers adopted graph-based methodologies to mine collective knowledge from massive GPS trajectories. However, GPS trajectories contain simple spatiotemporal information, which can hardly be used to discover tourists preference and seasonal attributes of POIs. Meanwhile, some recent works utilize geo-tagged photos [3], check-in data and reviews [4] separately to generate POIs recommendations. Although the LBSN data are suit for mining tourism semantics, they are not fit for generating travel sequences.

The flourish of the online travel websites provides an effective way to discover personalized travel sequences by mining massive user-generated travelogues. In recent works, researchers adopted travelogues and LBSN data to generate POIs sequences for individual tourist based on orienteering algorithm [5] and topic model methods [6] respectively.

In travel route recommending scenarios, researchers apply the sequential patterns mining methods to trajectory patterns mining and route recommending applications. Luo et al. [7] studied a new path finding system which discovers the most frequent path during user-specified time periods in large-scale historical trajectory data. Tsai et al. [8] proposed a touring path suggesting system for visitors to comprehend exhibits in exhibitions or museums. This system takes previous popular visiting trajectories as the suggestion foundation and provides a time-interval sequential patterns mining algorithm to generate personalized tours. However, all the above works cannot adopt tourism attributes, such as who is accompanying in trips and the visiting season of trips, to enhance the personalization of recommended POIs itineraries.

## 3 Research Methodologies

### 3.1 System Overview

The proposed system consists of three modules. The multi-source tourism big data integrating module aims to generate structured POIs visit sequences by incorporating

multifaceted tourism related information. The POI-Visit sequences mining module generates a series of POI-Visit sequential patterns, namely POIs travel routes, by adopting the POI-Visit PrefixSpan (PV-PrefixSpan) algorithm. The POIs travel routes recommendation module recommends optimal POIs itineraries to tourists under their personal constraints. Figure 1 illustrates the architecture of the proposed system.

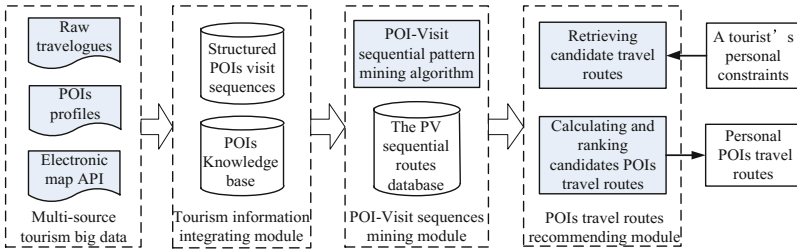


Fig. 1. The architecture of the POIs travel route recommendation system.

### 3.2 Multi-source Tourism Big Data Integrating Module POIs Knowledgebase Creation

In this work, the multi-source tourism includes real travelogues, POIs related profiles and POIs geographic information. To generate structured POIs visit sequences, a POIs knowledgebase need to be constructed first, which consists of a POIs attribute database and a POIs traffic transit matrix.

The POIs attribute database contains all of the POIs' attributes, including the system unique POI identifier (PID), the POI name, address and alias, ticket price, tourism types, visit duration, rating and suitable season etc. And we take Guilin city as an instance to construct the database. Table 1 shows an illustrative example of the partial POI attribute database. Each of POI in the database contains nine related attributes which are crawled and collated from the tourism related websites Ctrip<sup>1</sup> and Baidu Baike<sup>2</sup>.

To ensure the time accuracy of recommended travel routes, we use electronics map APIs<sup>3</sup> to acquire the traffic time of any pair of two POIs in the database, and store these times in the POIs traffic transit matrix. In detail, we utilize a reverse geocoding API to get the geographic location of a POI, and submit two POIs geographic locations to the road planning API to calculate the traffic time between the two POIs.

#### Structured POIs Visit Sequences Generation

In this work, the real travelogues are used to generate personalized POIs travel routes for various tourists. Therefore, we crawled massive real travelogues from Ctrip(see Footnote 1) to validate our system. The real travelogues are semi-structured data,

<sup>1</sup> <http://www.ctrip.com/>.

<sup>2</sup> <http://baike.baidu.com/>.

<sup>3</sup> <http://lbs.amap.com/api/>.



including POIs visit sequences and some tourism attributes, such as the visit season, the travel duration and the type of companion in the trip. The visit season records the month which the tourist made the trip. The type of companion records people who accompanied with the tourist in the trip, including child, parent, couple, friend and individual etc.

**Table 1.** An illustrative example of the partial POI attribute database.

PID	Name	Address	Rating	Duration	Price	Tourism types	Suitable season
P1	Elephantine Mountain Park	No. 1 Binjiang road, Guilin, China	4.1	3 h	¥70	Mountain; Natural Landscape; Cultural; Park;	Four seasons
P2	Reed Flute Cave	No. 1 Ludi road, Guilin, China	4.6	2 h	¥95	Cave; Rock landscape; Park;	Four seasons

As the travelogues probably contain noise information, a preprocessing method is proposed in this step. The noise information includes POI alias, or none-POI venues, such as hotels, streets, districts. For each of POI in a travelogue, the method converts a POI name to its PID by looking up the POIs attribute database, and inserts the corresponding POI visit duration and POI rating. The method deletes a POI from a sequence if its name is not matched. Next, the method inserts the traffic time of two consecutive POIs to generate the final structured POI visit sequence by looking up the POI traffic transit matrix. Finally, all generated sequences are divided into five subsets according to the specific companion type of the corresponding real travelogue, and stored in the structured POIs visit sequence database, abbr. SPSD. Note that, the structured POI visit sequence is defined as the POI-Visit (PV) pattern sequences.

**Definition 1.** A PV pattern  $\lambda_i$  is defined as  $\langle PI_i, PD_i, PR_i \rangle$  where  $PI_i$  is the POI identifier,  $PD_i$  is the visit duration,  $PR_i$  is the rating w.r.t POI  $i$ . The pattern  $\lambda_i$  is said to match the pattern  $\lambda_j$  if and only if  $PI_i = PI_j$ ,  $PD_i = PD_j$  and  $PR_i = PR_j$ .

**Definition 2.** Let  $A = \{\lambda_1, \lambda_2, \dots, \lambda_x\}$  be the PV patterns set and  $\delta_i$  be the traffic time between two POIs. A sequence  $\alpha = (V_1, \delta_1, V_2, \delta_2, \dots, \delta_{k-1}, V_k)$  is defined as a POI-Visit pattern sequence if  $V_s \in A$  for  $1 \leq s \leq k$ .

### 3.3 POI-Visit Sequences Mining Module

In this module, we designed a PV-PrefixSpan algorithm to mine PV sequential patterns, i.e. POIs travel routes, from the SPSD. Further, the proposed algorithm can generate more accurate POI routes in terms of time arrangement by handling the POI traffic time and visit duration separately. The algorithm related definitions are given.

**Definition 3.** Assume two PV sequences  $\alpha = (V_{\alpha 1}, \delta_{\alpha 1}, V_{\alpha 2}, \delta_{\alpha 2}, \dots, \delta_{\alpha(k-1)}, V_{\alpha k})$  and  $\beta = (V_{\beta 1}, \delta_{\beta 1}, V_{\beta 2}, \delta_{\beta 2}, \dots, \delta_{\beta(h-1)}, V_{\beta h})$  ( $h \leq k$ ),  $\beta$  is said to be contained in  $\alpha$  or a PV subsequence of  $\alpha$ , i.e.  $\beta \subseteq \alpha$ , if the sequence indexes  $1 \leq j_1 < j_2 < \dots < j_h \leq k$  exists such that, (1)  $V_{\beta 1} = V_{\alpha j_1}, V_{\beta 2} = V_{\alpha j_2}, \dots, V_{\beta h} = V_{\alpha j_h}$ ; (2)  $\delta_{\beta 1} = \delta_{\alpha j_1}, \delta_{\beta 2} = \delta_{\alpha j_2}, \dots, \delta_{\beta h} = \delta_{\alpha j_h}$ .

**Definition 4.** A PV sequence  $\alpha$  is called a POI-Visit (PV) sequential pattern if the number of sequences in the SPSD which contains  $\alpha$  as the subsequence is greater than or equal to the user specified minimum support, called  $min\_sup$ . A PV sequential pattern, which has  $l$  PV patterns, is denoted a  $l$ -length PV sequential pattern.

**Definition 5.** Given two PV sequential patterns  $\alpha = (V_{\alpha 1}, \delta_{\alpha 1}, V_{\alpha 2}, \delta_{\alpha 2}, \dots, \delta_{\alpha(k-1)}, V_{\alpha k})$  and  $\beta = (V_{\beta 1}, \delta_{\beta 1}, V_{\beta 2}, \delta_{\beta 2}, \dots, \delta_{\beta(h-1)}, V_{\beta h})$  ( $h \leq k$ ),  $\beta$  is a PV prefix of  $\alpha$  if and only if (1)  $V_{\beta i} = V_{\alpha i}$  for  $1 \leq i \leq h$ ; (2)  $\delta_{\beta i} = \delta_{\alpha i}$  for  $1 \leq i \leq h - 1$ .

**Definition 6.** Given two PV sequential patterns  $\alpha = (V_{\alpha 1}, \delta_{\alpha 1}, V_{\alpha 2}, \delta_{\alpha 2}, \dots, \delta_{\alpha(k-1)}, V_{\alpha k})$  and  $\beta = (V_{\beta 1}, \delta_{\beta 1}, V_{\beta 2}, \delta_{\beta 2}, \dots, \delta_{\beta(h-1)}, V_{\beta h})$  ( $h \leq k$ ),  $\beta$  is a subsequence of  $\alpha$ . Let  $1 \leq j_1 < j_2 < \dots < j_h \leq k$  be the sequence indexes of the PV sequential patterns contained in  $\alpha$  which match  $\beta$ . A subsequence  $\alpha' = (V_{\alpha' 1}, \delta_{\alpha' 1}, V_{\alpha' 2}, \delta_{\alpha' 2}, \dots, \delta_{\alpha'(g-1)}, V_{\alpha' g})$  of  $\alpha$ , where  $g = h + k - j_h$ , is named a projection of  $\alpha$  w.r.t.  $\beta$  if and only if (1)  $\beta$  is a PV prefix of  $\alpha'$  and (2) the last  $k - j_h$  PV patterns of  $\alpha'$  are same with the last  $k - j_h$  PV patterns of  $\alpha$ .

**Definition 7.** Let  $\alpha' = (V_{\alpha' 1}, \delta_{\alpha' 1}, V_{\alpha' 2}, \delta_{\alpha' 2}, \dots, \delta_{\alpha'(m-1)}, V_{\alpha' m})$  be the projection of  $\alpha$  w.r.t. a PV prefix  $\beta = (V_{\beta 1}, \delta_{\beta 1}, V_{\beta 2}, \delta_{\beta 2}, \dots, \delta_{\beta(h-1)}, V_{\beta h})$  ( $h \leq m$ ). Then  $\gamma = (V_{\alpha'(h+1)}, \delta_{\alpha'(h+1)}, V_{\alpha'(h+2)}, \delta_{\alpha'(h+2)}, \dots, \delta_{\alpha'(m-1)}, V_{\alpha'(m)})$  is the postfix of  $\alpha$  w.r.t. prefix  $\beta$ .

The goal of the algorithm is to discover all of potential frequent POIs travel routes in the SPSD. The frequent POIs travel routes are formalized as PV sequential patterns. As the original PrefixSpan [9] algorithm dose not includes the relationship among two PV patterns and their internal time, i.e. the POI traffic time, a PV\_Table is constructed to store this type of relation, where a column corresponds to a PV pattern and a row corresponds to a POI pair traffic time.

Concretely, the algorithm first uses each of 1-length PV sequential patterns to construct the corresponding  $\alpha$ -projection database from the SPSD, which is denoted as  $SPSD|_{\alpha}$  consisting of PV sequences w.r.t. the PV prefix  $\alpha$ . For each  $SPSD|_{\alpha}$ , the algorithm constructs the corresponding PV\_Table and recognizes every frequent table cell. The table cell records the support count of a specific PV pattern. Then, for each frequent cell appends the element  $(\delta_N, \lambda_j)$  to the end of  $\alpha$  to construct a longer PV sequential pattern  $\alpha'$ . Last the algorithm builds  $\alpha'$ -projection database  $SPSD|_{\alpha'}$ . Recursively constructing the PV sequential patterns in  $SPSD|_{\alpha'}$  discovers all of potential PV sequential patterns in the SPSD, which are stored in the frequent PV sequential pattern database, abbr. PVSPD.

### 3.4 POIs Travel Route Recommendation Module

Due to the PV sequential patterns are derived from the real travelogues, which contain rich tourism attributes and semantic information. Therefore, our module can utilize these attributes to match the input personal constraints and generate more personalized and reasonable POIs travel routes.

In our system, the input personal constraints include the travel duration, the type of companion in trip, the visit season and the preferring tourism types. This module first retrieves the candidate routes from the PVSPD according to the type of companion, and filters the candidate routes of which the total travel duration is equal or shorter than the input travel duration. During comparing duration, the module assumes that a tourist could spend 10 touring hours in each travel day. The total travel duration  $T_{DT}$  of a route is calculated by Eq. 1.

$$T_{DT} = \left( \sum_{i=1}^{|\alpha|-1} \delta_i + \sum_{i=1}^{|\alpha|} PD_i \right) \tag{1}$$

where  $|\alpha|$  is the length a POIs route  $\alpha$ ;  $\delta_i$  and  $PD_i$  are the  $i$ th traffic time and POI visit duration of  $\alpha$  respectively.

Next, the module adopts Eq. 2 to calculate the final rank score of each remaining candidate routes. The final rank score  $RScore_\alpha$  of the route  $\alpha$  derives from the tourism attributes, which consists of four parts, the total POI travel value  $TpValue_\alpha$ , the ratio of the total POI visit duration to the total route duration  $TdRatio_\alpha$ , the POI visit season value  $VsValue_\alpha$ , and the POI tourism type value  $TtValue_\alpha$ . These four parts are weighted by  $w_1, w_2, w_3$  and  $w_4$  respectively with the condition  $w_1 + w_2 + w_3 + w_4 = 1$ .

$$RScore_\alpha = (w_1 * TpValue_\alpha + w_2 * TdRatio_\alpha + w_3 * VsValue_\alpha + w_4 * TtValue_\alpha) \tag{2}$$

The  $TpValue_\alpha$  is the ranking of total POI ratings of route  $\alpha$ , which is calculated by Eq. 3.

$$TpValue_\alpha = \left( \sum_{i=1}^{|\alpha|} PR_i - \min(\{\sum_{i=1}^{|\alpha|} PR_i\}) \right) / \left( \max(\{\sum_{i=1}^{|\alpha|} PR_i\}) - \min(\{\{\sum_{i=1}^{|\alpha|} PR_i\}\}) \right) \tag{3}$$

where  $PR_i$  is the  $i$ th POI's rating of route  $\alpha$ ; and  $\{\sum_{i=1}^{|\alpha|} PR_i\}$  (with  $k = 1, \dots, m$ ) is the travel value set of the  $m$  retrieved candidate POIs travel routes; the function  $\min(*)$  and  $\max(*)$  select the minimum and maximum value from  $\{\sum_{i=1}^{|\alpha|} PR_i\}$  respectively.

The  $TdRatio_\alpha$  stands for the ratio of the total POI visit duration to the total duration of the route  $\alpha$ , which is calculated by Eq. 4.

$$TdRatio_\alpha = \sum_{i=1}^{|\alpha|} PD_i / T_{DT\alpha} \tag{4}$$

where  $\sum_{i=1}^{|\alpha|} PD_i$  is the summation of each POI visit duration in the route  $\alpha$ ;  $T_{DT\alpha}$  is the total travel duration of route  $\alpha$ .

The  $VsValue_\alpha$  represents the proportion of POIs in route  $\alpha$  that the POI's suitable season meets the input visit season constraint. This value is calculated by Eq. 5.

$$VsValue_\alpha = \left| \left\{ \sum_{i=1}^{|\alpha|} PI_i \mid \text{visit month} \in PI_i \text{ suitable season} \right\} \right| / |\alpha| \tag{5}$$

where  $\left| \left\{ \sum_{i=1}^{|\alpha|} PI_i \mid \text{visit month} \in PI_i \text{ suitable season} \right\} \right|$  is the count of POIs subset of which the POI's suitable season matches the user input visit season.

The  $TtValue_\alpha$  represents the proportion of POIs in the route  $\alpha$  that the POI's tourism types meets the input preferring tourism types. This value is calculated by Eq. 6.

$$TtValue_\alpha = \left| \left\{ \sum_{i=1}^{|\alpha|} PI_i | \text{preferring types} \in PI_i \text{ types} \right\} \right| / |\alpha| \quad (6)$$

where  $|\{\sum_{i=1}^{|\alpha|} PI_i | \text{preferring types} \in PI_i \text{ types}\}|$  is the count of POIs contained in route  $\alpha$  if the POI's tourism types meet the user input preferring tourism types.

## 4 Experiment and Discussion

### 4.1 Data Set and Experiment Settings

The structured POIs visit sequences are generated from 10,109 Guilin related real travelogues. In detail, there are 5,694 structured POIs visit sequences included in the sequences set, 132 POIs stored in the POIs attribute database, and 8,646 traffic times of POI pairs included in the POI traffic transit matrix.

In the following sections, the minimum support is set at 0.4%; the recommending weight  $w_1$  is set at 0.4, and the weights  $w_2$ ,  $w_3$  and  $w_4$ , are equally set at 0.2.

### 4.2 Validation Experiment

In this section, we choose two popular baseline methods to demonstrate advantages of our system in recommending rich tourism attributes POIs travel routes. The Random Selecting Method [5] randomly selects a series of POIs from the POI attribute database to generate a POIs route. The Popularity Ranking Method [10] constructs a POIs route by selecting POIs from the POI popularity list according a descending ordering. The POI popularity is derived from the visit quantity of a specific POI in the POI-Visit sequences dataset. The baseline procedure stops when the total travel duration of the generating route reaches the input trip duration constraint.

To validate the effectiveness of our method compared to the baseline methods, we assume a tourist who intends to make a 1 day trip with his or her spouse in summer, and prefers cultural, shopping, river and natural landscape type of POIs. The results of our method and two baseline methods are shown in Table 2. All recommended routes meet the trip duration constraint. Under these constraints, our method retrieves 1,337 candidate POIs routes from the couple type PVSPD, while the popularity-based method generates only one fixed POI routes. In detail, our system recommends a 9 h POIs travel route which including 8 h total POI visit duration and 1 h total transit time, and possessing 13.4 points of the total POI rating, all the four preferred tourism types are met by the recommended POIs. The Popularity Ranking Method generates a 13.3 points POI rating route, however, the total POI transit time is 4 h, which is difficult for a tourist to finish this route in a single day. Further, this method is lack of routes diversity due to the fixed result. Although the random selecting method can generate massive candidate POIs routes, it cannot ensure the other tourism attributes of the final recommended routes.

Table 3 summarizes the advantages of our method compared to the baseline methods. Both of the two baseline methods cannot recommend a reasonable POIs route by considering POIs visit sequence, however, our method uses real travelogues to generate POIs routes which ensures a rational POI visit order, i.e. keeps shorter traffic time routes. Our method exhibits effective performance in recommending POIs routes by considering rich tourism attributes contained in travelogues and POIs knowledge-base, such as the type of companion in trip, the visit season and the preferring tourism types.

**Table 2.** Results of POIs travel routes recommended by three methods.

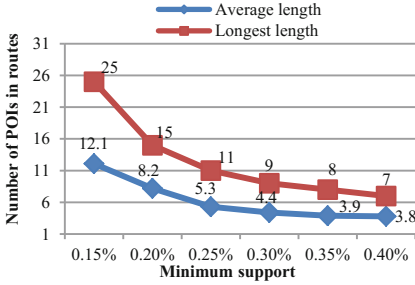
Methods	Recommended POIs travel routes	Total POI rating	Transit time/visit duration	Matched tourism types
Our method	Li River->Yangshuo West Street->Impression Sanjie Liu	13.4	1 h/8 h	4
Pop. rank	Yangshuo West Street->Li River->Yulong River	13.3	4 h/7.5 h	3
Rand. select	West Mountain Park->Gudong Waterfall->Nanxi Park	11.5	3.5 h/7 h	2

**Table 3.** Comparison of three recommending methods.

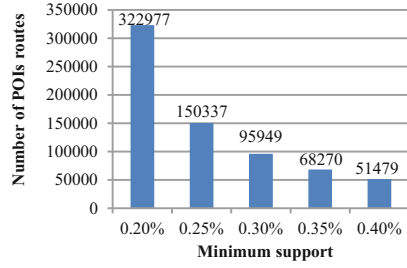
	Trip duration constraint	Diversity of routes	Travel value of routes	Rationality of routes	Tourism attributes of routes
Our method	Good	Good	Good	Good	Good
Pop. rank	Good	Bad	Good	Bad	Bad
Rand. select	Good	Good	Bad	Bad	Bad

### 4.3 Performance Experiment

Figure 2 illustrates the performance of our algorithm, i.e. the quality and the quantity of the generated POIs travel routes, under various *min\_sup* settings. These experiment results demonstrate the proposed algorithm can effectively generate diverse POIs travel routes while have rather high travel quality.



(1) The quality of POIs routes



(2) The quantity of POIs routes

Fig. 2. The performance of algorithm under various *min\_sup*.

### 5 Conclusions and Future Work

In this work, we present the multi-source tourism big data integrating method to construct POIs visit sequences by using travelogues, POIs profiles and geographic information. And the POI-Visit pattern sequence mining algorithm is proposed to generate diverse personalization POIs travel routes. Finally, the POIs travel route recommendation method is designed to recommend a list of personalization POIs travel routes by considering tourist constraints and tourism attributes. The experiments on real dataset show that the proposed system is efficient and effective in recommending personalization POIs travel routes. In the future, we intend to extend our system to incorporate more types of tourism resources, such as hotels, restaurants and shopping venues, to recommend integrated travel routes.

**Acknowledgement.** This work was partially supported by the National Natural Science Foundation of China (Nos. U1501252, 61572146, U1711263), the Natural Science Foundation of Guangxi Province (No. 2016GXNSFDA380006), the Guangxi Innovation-Driven Development Project (No. AA17202024), and the Innovation Project of GUET Graduate Education (Nos. 2018YJXC12, YCSW2018139).

### References

1. Zhang, C., Wang, K.: POI recommendation through cross region collaborative filtering. *Knowl. Inf. Syst.* **46**(2), 369–387 (2016)
2. Xiao, X., Zheng, Y., Luo, Q., Xie, X.: Finding similar users using category-based location history. In: *Proceedings of the 18th Annual ACM International Conference on Advances in Geographic Information Systems*, pp. 442–445. ACM (2010)
3. Bhargava, P., Phan, T., Zhou, J., Lee, J.: Who, what, when, and where: multi-dimensional collaborative recommendations using tensor factorization on sparse user-generated data. In: *WWW 2015*, pp. 130–140 (2015)
4. Xie, M., Yin, H., Wang, H., Xu, F., Chen, W., Wang, S.: Learning graph-based POI embedding for location-based recommendation. In: *CIKM 2016*, pp. 15–24 (2016)

5. Lim, K., Chan, J., Leckie, C., Karunasekera, S.: Personalized tour recommendation based on user interests and points of interest visit durations. In: IJCAI 2015, pp. 1778–1784 (2015)
6. Jiang, S., Qian, X., Mei, T., Fu, Y.: Personalized travel sequence recommendation on multi-source big social media. *IEEE Trans. Big Data* **2**(1), 43–56 (2016)
7. Luo, W., Tan, H., Chen, L., Ni, L.M.: Finding time period-based most frequent path in big trajectory data. In: SIGMOD Conference 2013, pp. 713–724 (2013)
8. Tsai, C., Liou, J.J.H., Chen, C., Hsiao, C.: Generating touring path suggestions using time-interval sequential pattern mining. *Expert Syst. Appl.* **39**(3), 3593–3602 (2012)
9. Pei, J., Han, J., Mortazavi-Asl, B., Pinto, H., Chen, Q., Dayal, U., Hsu, M.: PrefixSpan: mining sequential patterns by prefix-projected growth. In: ICDE 2001, pp. 215–224 (2001)
10. Cai, G., Lee, K., Lee, I.: Itinerary recommender system with semantic trajectory pattern mining from geo-tagged photos. *Expert Syst. Appl.* **94**, 32–40 (2018)



# Analysing TV Audience Engagement via Twitter: Incremental Segment-Level Opinion Mining of Second Screen Tweets

Gavin Katz, Bradford Heap, Wayne Wobcke<sup>(✉)</sup>, Michael Bain,  
and Sandeepa Kannangara

School of Computer Science and Engineering,  
University of New South Wales, Sydney, NSW 2052, Australia  
gavinkatz@gmail.com, {b.heap,w.wobcke,m.bain,s.kannangara}@unsw.edu.au

**Abstract.** To attract and retain a new demographic of viewers, television producers have aimed to engage audiences through the “second screen” via social media. This paper concerns the use of Twitter during live television broadcasts of a panel show, the Australian Broadcasting Corporation’s political and current affairs show Q&A, where the TV audience can post tweets, some of which appear in a tickertape on the TV screen and are broadcast to all viewers. We present a method for aggregating audience opinions expressed via Twitter that could be used for live feedback after each segment of the show. We investigate segment classification models in the incremental setting, and use a combination of domain-specific and general training data for sentiment analysis. The aggregated analysis can be used to determine polarizing and volatile panellists, controversial topics and bias in the selection of tweets for on-screen display.

**Keywords:** Social TV · Opinion mining · Machine learning  
Social media

## 1 Introduction

In order to attract a new demographic of viewers, traditional broadcast media have explored ways to embrace new technologies to provide a more engaging and interactive TV viewing experience. A common approach is to augment a broadcast television show through the use of social media channels. This means of audience participation is known as the “second screen” [7]. Common uses of the second screen include voting for contestants on game shows or reality TV, posting messages about a show or the current episode, and communicating with cast members as the show airs. Previous computational work on “second screening” [3, 8] has studied sentiment in human annotated datasets [1], time series analysis over the frequency of tweets and human coded content analysis [3], and the interaction between Twitter users [5].



This paper presents a method to analyse audience opinion via the second screen during live television broadcasts of a panel show, the Australian Broadcasting Corporation’s weekly political and current affairs show Q&A. During a live broadcast of the program, viewers are encouraged to engage with what is being discussed on the show by posting tweets containing the hashtag #QandA, some of which are chosen for display on a tickertape on the bottom of the screen. This mechanism enables feedback, directed towards both topical segments and panellists, to be given in real time, potentially enabling the host to comment on the feedback during the show, or even adjust the content of the show in response to viewer feedback.

In this paper, we show how the model could be used for aggregation of opinions over entities and across segments, to address the questions of identifying polarizing panellists, controversial topics and bias in the selection of tweets for display. We investigate *incremental* opinion mining methods that compute aggregated audience opinions at the end of each segment of an episode. The model classifies audience tweets into two components: the *segment* (part of the show the tweet is about) and the *target* (a panellist or person or organization discussed in the show). Each segment is defined by a question from the studio audience, usually on a topical political issue, which is then discussed by the panellists. There are around 7–8 segments in a typical episode, which runs for 65 min. Issues of computational efficiency and training data are paramount. Importantly, the show is not pre-scripted and questions are not announced in advance, and moreover, there is little overlap between questions from week to week, so it is impossible to use previous episodes as training data for segment classification. Our key idea is to use the episode transcript itself – which is available in real time – as training data for incremental models. We investigate classification of tweets into segments using Multinomial Naïve Bayes (MNB) and Support Vector Machines (SVM) in an incremental setting where models are retrained and run at the end of each segment, using the transcript up to that point, to classify the tweets posted during that segment.

For evaluation, we selected three episodes of Q&A – Episode 1: August 1, 2016 (8 segments); Episode 2: September 12, 2016 (8 segments); Episode 3: September 19, 2016 (7 segments). The dataset consists of all tweets containing the #QandA hashtag posted across the live broadcasts from 9.30pm (the start of the show) until 11.00pm AEST, i.e. until around 25 min after the end of the episode, and contains 17,044 tweets from 3,696 distinct authors for Episode 1, 17,274 tweets from 3,029 distinct authors for Episode 2, and 15,983 tweets from 3,765 distinct authors for Episode 3. A ground truth dataset was defined by manually labelling a random sample of 1,659 distinct tweets posted during the live broadcasts of the three episodes. A tweet is categorized into a segment if it references any part of the discussion aired, or is specifically targeting any of the themes raised in the discussion. Otherwise, if the tweet is relevant to the episode but not to any specific segment, it is labelled under a *General* category (300 tweets in the ground truth dataset).

## 2 Incremental Segment Classification

To explain the idea behind the incremental segment classification models, let the segments of an episode be  $S_1, S_2, \dots$ . Then, at the end of each segment  $S_n$  (starting with  $S_3$ , when there is sufficient training data), a new model  $M_n$  is built by training on all the transcript text (ignoring annotations) up to the end of  $S_n$ . Thus models  $M_3, M_4, \dots$  are created up to the end of the episode. The models are used incrementally to classify the tweets posted in each new segment: more precisely,  $M_n$  is always used to classify the tweets posted during  $S_n$ , but note that these tweets could be classified as relating to any of the segments  $S_1, \dots, S_n$  (though not subsequent segments). Equivalently, we can think of the “incremental” model  $I_n$ , computed at the end of segment  $S_n$ , as a model that uses the time a tweet was posted to “select” the model  $M_i$  ( $i \leq n$ ) to classify the tweet into one of the segments  $S_1, \dots, S_n$ .

Use of an incremental model requires a base model for the classification of tweets posted during  $S_n$  into the classes  $S_1, \dots, S_n$ . We evaluate two commonly used supervised classification models in this setting: Multinomial Naïve Bayes (MNB) [6] and Support Vector Machines (SVM) [4], using Weka’s<sup>1</sup> implementation of the Sequential Minimal Optimization (SMO) training algorithm. These algorithms have previously been shown to train quickly and perform well on limited training datasets [2], thus are suitable for use in the context of the show to train and run the classifiers at the end of each segment. To maximize classification accuracy, each algorithm has a tuned text pre-processing stage, and to maximize each classifier’s F1 measure, threshold functions are defined for each method, acting as a “confidence level” for a given tweet to be classified: any tweet whose confidence is below the threshold is assigned the *General* class.

Pre-processing techniques were developed for the entire system to use, and then a pre-processing pipeline was chosen for each algorithm that was tuned to maximize the F1 measure on training data. For the training data, all punctuation and annotations were removed and text was reduced to lower case. This results in a transcript that contains only the exact words spoken on the show. Tweets were also subject to two additional pre-processing steps: (1) any mention of an account handle (i.e. word starting with “@”) is converted to the full name for that account; and (2) hashtags that run together separate words beginning with a capital letter are split into several words (e.g. the hashtag #KevinRudd is converted to “Kevin Rudd”). Finally, for MNB (but not SVM), stop words are removed and words lemmatized, as is standard practice for achieving the best results with these models on text classification problems.

### 2.1 Segment Classification Evaluation

Incremental segment classification is evaluated using the three episode ground truth dataset described above. MNB and SVM classification models are used as the base models and trained incrementally on the transcript data for these

<sup>1</sup> <http://www.cs.waikato.ac.nz/ml/weka/>.

episodes. To emphasize that the segment-based incremental model is being used, we call the incremental models I-MNB and I-SVM. These models are compared to a baseline classifier that is a simple time-based classifier where each tweet posted during a segment is assigned to that same segment class, i.e. simply assumes each tweet is relevant to the currently airing segment, as in Diakopoulos and Shamma [1].

Table 1 shows the precision, recall and F1 measure for all tweets in the ground truth dataset for each of the three episodes. These results show that the I-MNB classifier is consistently the best model for classifying tweets into segments. The simple baseline works surprisingly well (though of course fails to capture the “lag” between Twitter stream and show), and even outperforms I-SVM on Episodes 1 and 2. Hence all further analysis in this paper is done with I-MNB.

**Table 1.** Episode segment classification

	Episode 1			Episode 2			Episode 3		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
Baseline	0.69	<b>0.82</b>	0.75	0.63	0.77	0.69	0.49	0.59	0.54
I-MNB	<b>0.80</b>	0.79	<b>0.79</b>	<b>0.78</b>	<b>0.79</b>	<b>0.79</b>	<b>0.68</b>	<b>0.70</b>	<b>0.69</b>
I-SVM	0.59	0.61	0.60	0.60	0.65	0.63	0.52	0.59	0.55

Table 2 shows the per-segment precision, recall and F1 for each of the segment classes in Episode 3, as calculated at the end of the episode, i.e. using  $I_7$ , the model that uses  $M_i$  to classify the tweets posted during the segment  $S_i$  into  $S_1, \dots, S_i$  (and  $M_3$  to classify tweets posted during  $S_1, S_2$  and  $S_3$ ). Segment 2 (*Plebiscite and Suicide*) has an outlying (low) F1 measure. In this case, the only other question discussed previously in this episode is the first segment, *Plebiscite or Wait*. As there is a drop in F1 measures (from 0.63 to 0.53) between Segment 1 and Segment 2, this suggests that the  $M_3$  classifier, which has to decide between the first three segments, heavily favours the *Plebiscite or Wait* segment: the words overlap between the two segments, and for MNB the length of the training text has a large effect on the behaviour of the classifier. In particular, *Plebiscite or Wait* has a length of 1,136 words, significantly higher than both the average length and the *Plebiscite and Suicide* length of 700 words.

**Table 2.** Episode 3 - end of episode analysis, MNB

	$S_1$ Plebiscite or wait	$S_2$ Plebiscite and suicide	$S_3$ Hanson speech	$S_4$ Asylum	$S_5$ Migrants today	$S_6$ Alcohol abuse	$S_7$ Creative copyright
Precision	0.52	0.64	0.80	0.81	0.61	0.88	0.87
Recall	0.79	0.46	0.75	0.79	0.75	0.59	0.72
F1	0.63	0.53	0.78	0.80	0.67	0.71	0.79

**Table 3.** Episode 3 - end of segment analysis, F1 measure

	$S_3$ Hanson speech	$S_4$ Asylum	$S_5$ Migrants today	$S_6$ Alcohol abuse	$S_7$ Creative copyright
Baseline	0.46	0.43	0.46	0.60	0.63
I-MNB	0.88	0.77	0.77	0.73	0.79

Table 3 shows the F1 measure for each segment of Episode 3, calculated at the end of the segment, comparing the score for only that class on tweets posted during that segment for the two classifiers, the MNB model constructed at the end of the segment and the baseline time-based classifier. This is the scenario that arises if those scores were to be calculated during the episode for the most recently aired segment. Immediately noticeable is the very poor performance of the baseline, indicating that in this episode much of the Twitter discussion relating to a segment spills over into following segments. The results for I-MNB are consistent with Table 2, showing that the method can be used incrementally in the desired fashion. F1 is expected to be higher in the per-segment calculations because only F1 for the current class is reported. Taken as a whole, this indicates that I-MNB could be used during the episode for classification of tweets into segments during the show.

### 3 Aggregated Audience Opinion

We now consider the main purpose of the “second screen” analysis, to address *aggregated* audience opinion, in a mode that, again, could be presented during a show at the end of each segment in a live broadcast. Sentiment analysis was done using SVM trained on a combined general and Q&A specific corpus (for each episode, trained using a general Twitter sentiment analysis corpus<sup>2</sup> and the sentiment-tagged tweets from the other two episodes), which, consistent with prior research, provided the best results, with precision 59%, recall 56% and an F1 measure of 58%.

Specific questions of interest are: (i) which panellists are polarizing, (ii) which panellists are “volatile” in the sense that sentiment towards them varies according to the segment, (iii) which topics are most controversial, and (iv) whether there is any bias in the selection of tweets for on-screen display. This relies on a way to aggregate audience tweets – this is done by identifying the explicit target of each tweet and then clustering tweets with the same target, as determined using single-linkage clustering based on the Sørensen-Dice coefficient with a similarity threshold of 0.15. An aggregated sentiment for an entity cluster or a segment is calculated by taking the mean sentiment over a collection of tweets, where *positive* is 1, *neutral* is 0, and *negative* is  $-1$ .

Entity recognition uses two standard tools to identify entities within tweets: Basis Technology’s Rosette Entity Extractor (REX) and Stanford CoreNLP

<sup>2</sup> <http://www.mpi-inf.mpg.de/~smukherjee/data/twitter-data.tar.gz>.

Named Entity Recognition (NER), run with their default settings. If a tweet includes multiple entities, the most important entity is assumed to be the target, and is determined heuristically using a hierarchy of entity types: (i) person, organization, (ii) location, nationality, religion, (iii) title, and (iv) all other types.

### 3.1 Polarizing and Volatile Panellists

Table 4 shows the number of tweets in the whole Q&A dataset labelled by each of the three methods, and the total number of complete tuples, where a complete tuple is of the form  $\langle e_i, c_{ij}, s_{ij} \rangle$  with an entity, segment and sentiment. The methods used are incremental MNB for segment classification and SVM for sentiment classification.

**Table 4.** Tweets labelled for segment, sentiment and entities

Episode	Tweets	Segment classified	Sentiment classified	Entities recognized	Complete tuples
1	17,044	13,933	16,286	11,586	9,195
2	17,274	13,018	16,529	12,588	9,107
3	15,983	13,343	15,238	12,061	9,708

For each episode, more than 50% of tweets are assigned a complete tuple that can be used for opinion aggregation by segment and entity. However, since a tweet is labelled with the *General* category if the confidence for classifying it into any specific segment falls below a threshold, a large number of tweets containing entities are not assigned to any segment. In this case, the tweet is almost certainly directly about an entity, often a panellist but sometimes also the subject of the discussion. We use tweets of this form to assess aggregated opinion towards entities across an entire episode.

We calculate the mean absolute deviation (MAD) of the sentiments, and the standard deviation and 95% confidence interval (CI) around the mean sentiment. These measures are used to determine the degree of polarity in the tweets that contribute to the mean sentiment. A large MAD, standard deviation and confidence interval suggests that the authors of tweets have divergent views, while a small MAD and standard deviation suggests that the majority of tweeters are in agreement on the sentiment.

Table 5 shows some sample results of the aggregated opinion mining on tweets in the *General* category from the three episodes. All four entity clusters shown in this table correspond to panellists on the episodes: Matt Canavan,<sup>3</sup> Jimmy Barnes, Magda Szubanski<sup>4</sup> and Jacqui Lambie. Across all of these entity clusters,

<sup>3</sup> Resources minister in the government.

<sup>4</sup> Lesbian actor/comedian, strong supporter of same sex marriage.

**Table 5.** Aggregated opinion – general segment classification

Entity cluster	Tweets	Mean sentiment	MAD	Std deviation	95% CI
MATT CANAVAN, ...	124	-0.46	0.73	0.83	-0.61 to -0.31
JIMMY BARNS, JIMMY, ...	130	-0.34	0.76	0.84	-0.48 to -0.20
YAY MAGDA, MAGDA, ...	89	0.17	0.77	0.85	-0.01 to 0.35
JACQUIE LAMBIE, ...	82	-0.07	0.81	0.89	-0.26 to 0.12

**Table 6.** Aggregated opinion – specific segment classification

Entity	Tweets	Mean sentiment	MAD	Std deviation	95% CI
Matt Canavan	186	-0.40	0.76	0.84	-0.52 to -0.28
Jimmy Barnes	96	-0.77	0.40	0.60	-0.89 to -0.65

there is a similar high level of standard deviation, indicating a wide variety of sentiment.

As Q&A aims to discuss topical controversial issues, it is unsurprising that much of the sentiment is negative; the confidence intervals for the first three panellists in the table are all clearly in the negative. On the other hand, the entity cluster for Magda Szubanski, with a mean sentiment of 0.17, is highly unusual, being the only entity cluster studied with a positive skew. The last entity cluster for Jacqui Lambie is also unusual in having a high MAD and standard deviation and a neutral mean sentiment. From this it can be concluded that Jacqui Lambie is more polarizing than other panellists, provoking a wider range of sentiment from the TV audience.

Our method of opinion mining also allows tweets to be aggregated by entity within segments. Table 6 shows the results for two of the same panellists, Matt Canavan and Jimmy Barnes, during specific (different) segments when they were active participants in the discussion. What is noticeable is that all values for Matt Canavan are virtually identical to those in Table 5, indicating that sentiment towards him during the segment is representative of overall sentiment. On the other hand, sentiment towards Jimmy Barnes is more volatile, in that sentiment is much more negative towards him during this segment than overall: the mean sentiment in this table (-0.77) falls well outside the confidence interval in Table 5, in support of this conclusion.

### 3.2 Controversial Segments

Our methods can be used to aggregate sentiment over all tweets classified as belonging to a segment, and to determine controversial topics, we focus on outliers on these metrics. Note that for most segments, the values tend to follow the trends for overall sentiment of Q&A tweets discussed above, so are not revealing. One outlier on standard deviation is the segment *Gun Laws and Terrorism* which has standard deviation of 0.88, indicating an even more highly controversial topic than usual.

**Table 7.** Aggregated opinion – specific segment classification

Entity	Tweets	Mean sentiment	MAD	Std deviation	95% CI
<i>On-screen tweets</i>					
Kevin Rudd	6	-0.67	0.44	0.47	-1 to -0.29
Tony Abbott	5	-0.6	0.64	0.89	-1 to 0.18
Jacqui Lambie	2	1	0	0	1 to 1
<i>Q&amp;A tweets</i>					
KRUDD, RUDDS, ...	600	-0.49	0.67	0.76	-0.55 to -0.43
TONY ABBOT, ...	301	-0.54	0.65	0.77	-0.63 to -0.46
JACQUIE LAMBIE, ...	180	-0.03	0.84	0.91	-0.16 to 0.10

### 3.3 Bias in Selection of Broadcast Tweets

As part of the broadcast of Q&A, between 12–20 tweets are chosen manually for on-screen display during the panel discussion. Table 7 shows three examples of people about whom tweets were displayed (in different segments) and who provoked a large number of audience tweets: Kevin Rudd,<sup>5</sup> Tony Abbott<sup>6</sup> and Jacqui Lambie. Only the last of these was a panellist. The confidence intervals for the on-screen tweets for the first two are quite wide (-1 to -0.29 for Rudd; -1 to 0.18 for Abbott), and the mean sentiment derived from the Q&A tweets falls within these ranges. Moreover, the confidence intervals for the Q&A tweets are completely contained within those for the on-screen tweets. Lambie is an anomaly, with only two tweets shown, both positive, whereas, as noted before, she is a polarizing panellist.

## 4 Conclusion

In this work, we have presented a method for computing the aggregated opinion of Twitter users towards entities within segments discussed on the popular Australian current affairs panel television show Q&A. The idea is that aggregated opinions from the “second screen” could be presented during the show at the end of each segment, and used to provide further feedback to the TV audience.

The key insight is to develop an incremental, segment-level, opinion mining model that can be trained on the transcript of the episode. We used an incremental version of Multinomial Naïve Bayes for classification of tweets into segments, and Support Vector Machines trained on a combination of a general Twitter sentiment corpus and specific Q&A tweets for sentiment classification. We showed how these techniques can be used to address the questions of which panellists most polarize the audience, which panellist’s sentiments fluctuate according to the topical segment, which topics are most controversial, and whether there is bias in the selection of tweets for on-screen display.

<sup>5</sup> Former Labor Australian Prime Minister.

<sup>6</sup> Former Liberal Australian Prime Minister.

**Acknowledgement.** Thanks to Data to Decisions Cooperative Research Centre for supporting this research and supplying full access to the Twitter data for this paper.

## References

1. Diakopoulos, N., Shamma, D.A.: Characterizing debate performance via aggregated Twitter sentiment. In: Proceedings of the 28th ACM Conference on Human Factors in Computing Systems, pp. 1195–1198 (2010)
2. Forman, G., Cohen, I.: Learning from Little: Comparison of classifiers given little training. In: Boulicaut, J.-F., Esposito, F., Giannotti, F., Pedreschi, D. (eds.): PKDD 2004. LNCS (LNAI), vol. 3202. Springer, Heidelberg (2004). <https://doi.org/10.1007/b100704>
3. Giglietto, F., Selva, D.: Second screen and participation: a content analysis on a full season dataset of tweets. *J. Commun.* **64**, 260–277 (2014)
4. Joachims, T.: Text categorization with Support Vector Machines: learning with many relevant features. In: Nédellec, C., Rouveirol, C. (eds.) ECML 1998. LNCS, vol. 1398, pp. 137–142. Springer, Heidelberg (1998). <https://doi.org/10.1007/BFb0026683>
5. Lin, Y.R., Keegan, B., Margolin, D., Lazer, D.: Rising tides or rising stars?: Dynamics of shared attention on Twitter during media events. *PloS ONE* **9**(5), e94093 (2014)
6. McCallum, A., Nigam, K.: A comparison of event models for Naive Bayes text classification. In: Sahami, M. (ed.) Learning for Text Categorization: Papers from the AAAI Workshop. AAAI Press, Menlo Park, CA (1998)
7. Proulx, M., Shepatin, S.: Social TV. Wiley, Hoboken, NJ (2012)
8. Vaccari, C., Chadwick, A., O’Loughlin, B.: Dual screening the political: media events, social media, and citizen engagement. *J. Commun.* **65**, 1041–1061 (2015)





# Absolute Orientation and Localization Estimation from an Omnidirectional Image

Ruyu Liu, Jianhua Zhang<sup>(✉)</sup>, Kejie Yin, Zhiyin Pan, Ruihao Lin, and Shengyong Chen

Zhejiang University of Technology, Hangzhou, Zhejiang, China  
zjh@zjut.edu.cn

**Abstract.** In this paper, we present a novel method for instantly and autonomously determining global-localization and pose in a wide-area outdoor environment based on a single panoramic image and a 2.5D city model. In contrast to existing method, our approach is not entirely dependent on prior GPS data and inclined to use omnidirectional visual information to provide a precise city-localization in urban scene.

We estimate the orientation and localization of camera based on spherical panoramic imaging model. We evaluate the proposed method on a challenging dataset. The experiments indicate that pose precision from our method is obviously superior to that from the consumer sensors and we remain unbeatable in terms of time cost compared to previous methods.

**Keywords:** Pose estimation · Omnidirectional vision  
Geo-localization

## 1 Introduction

Absolute camera position and pose estimation plays a crucial role in computer vision and visual robot navigation. Urban outdoor localization typically depends on sensor equipments, such as the Global Positioning System (GPS) [10] and inertial measurements units (IMUs). However, there are inevitable drawbacks in GPS and IMU. For example, the accuracy of consumer GPS is not enough for many applications, such as outdoor Augmented Reality (AR), and it suffers from substantial errors in the urban canyons. The disadvantages of the IMU is its error accumulates over time, which can cause large pose errors.

Vision-based localization approach is a promising alternative. However, it relies on the pre-captured image database [7] or pre-built point cloud models [9] that registered offline, which is time-consuming and does not scale well. Later work [4, 8] avoid image databases or point clouds by using untextured models. Taneja *et al.* [8] provide an iterative optimization method for correcting the prior panoramic image pose with comparably detailed 3D models. Other methods like

visual SLAM [6] and VO only provide the relative pose in an arbitrary coordinate with an unknown scale.

In this paper, we introduce a novel approach for instantaneously estimating absolute orientation and global 3D position in large-scale urban scene. There are two essential stages in our approach. Firstly, the absolute camera orientation (roll, pitch, yaw) is instantly estimated based on geometric information from a single panoramic image. Secondly, the absolute 3D position is then calculated by registering the panorama to a 2.5D city map. The 2.5D map which merely consists of 2D building footprints and approximate building heights, can easily be obtained from map software, such as OpenStreetMap<sup>1</sup>. Moreover, the pretty accurate pose obtained by our method can satisfy the requirement of following tasks, such as outdoor AR.

Maybe the most relevant work to ours are Bazin *et al.* [2] and Arth *et al.* [1]. In [2], authors propose a method for estimating rotation and vanishing points by the omnidirectional vision in urban environment. However, the estimated 3-DOF pose is relative and there is little information available about absolute localization estimation. [1] estimates global 6-DOF pose from the 2.5D model. But the limitations in [1] is that this method only focuses on the narrow field-of-view and the time cost is high. In contrast, our approach takes advantage of omnidirectional vision to obtain more details of scene information.

Our contributions can be summarized as follows:

- (1) A framework for the global localization and 7-DOF (orientation, position, scale) absolute pose estimation with a 2.5D map. Additionally, the registration method is sensor-independent to a certain extent.
- (2) The omnidirectional vision used in pose estimation can acquire full 360-degree field-of-view (FOV) scene information, which is beneficial to the accuracy of orientation estimation and improves the visual interactive experience in wide-area outdoor environment.
- (3) Our method develops a common localization technique, registers a single panoramic image with respect to the untextured model by matching building outlines, and replaces the redundant point cloud models or edge models of the environment with 2.5D maps.

## 2 Approach

The objective of our method is to instantly and accurately achieve the globally aligned pose and 3D geo-localization estimation in absolute coordination system.

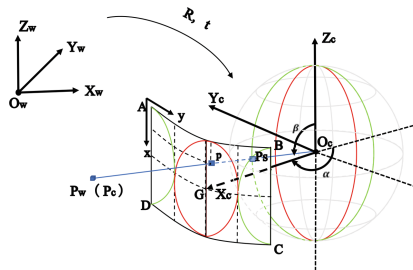
### 2.1 Spherical Panoramic Imaging Model

The panoramic image can provide more geometric structure information compared to the traditional camera. As a convention, we use right upper case superscription to indicate the coordinate systems, and use the bold and Italic to

<sup>1</sup> OpenStreetMap: <http://www.openstreetmap.org>.

represent the vector. As shown in Fig. 1, let us define three coordinate systems as follows.

- The image coordinate system is the 2D plane coordinate system which takes  $\mathbf{A}$  as the origin, the  $x$ -axis is pointing to  $\mathbf{D}$ , the  $y$ -axis is pointing to  $\mathbf{B}$  from  $\mathbf{A}$ . The width of panorama  $W$  is  $W = 2\pi r$ , the height  $H$  is  $H = \pi r$ .  $\mathbf{p} = (x, y)$  represents a 2D pixel in the panoramic image coordinate system.
- The spherical camera coordinate system is the 3D coordinate system which takes the center point of sphere  $\mathbf{O}_c$  as the origin and the radius of panoramic sphere is  $r$ .  $\mathbf{P}_c = (X_c, Y_c, Z_c)$  is a space point in the camera coordinate system.  $\mathbf{P}_s = (\alpha, \beta, r)$  represents a 3D point on the sphere surface where  $\alpha$  is longitude,  $\beta$  is latitude.  $r$  is the radius of sphere. The  $x$ -axis is pointing to  $\mathbf{G}$  from the  $\mathbf{O}_c$ , the  $z$ -axis is pointing vertically, the  $y$ -axis is perpendicular to the  $x$ -axis and  $z$ -axis.
- World coordinate system is a 3D coordinate system whose  $y$ -axis is pointing to the due north, the  $x$ -axis is pointing to the due east and the  $z$ -axis is pointing to the sky vertically. Let the  $\mathbf{O}_w$  as the world coordinate origin. A space point  $\mathbf{P}_w$  in the absolute world coordinate system is denoted as  $\mathbf{P}_w = (X_w, Y_w, Z_w)$ .



**Fig. 1.** The mapping relationship between spherical panoramic image and plane panoramic image using the spherical camera model

## 2.2 Estimating Orientation

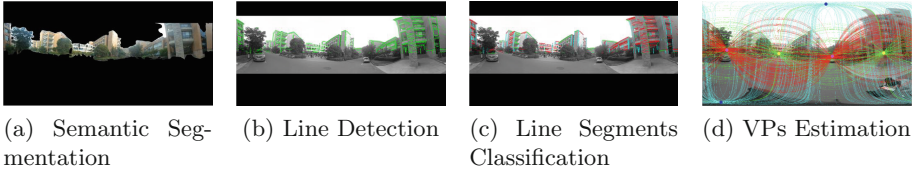
The procedure of computing the orientation can be divided into two parts: estimating the vertical axis (the roll and pitch) and orientation of vertical axis.

**Estimating Vertical Axis.** In order to estimate the rotation of vertical axis, we calculate vanishing points by the visual information, followed by three steps:

**Semantic Segmentation.** We first perform a pixel-wise semantic segmentation for a given panoramic image using FCN algorithm [5]. By doing so, the background of the panorama *i.e.*, no-building segment is separated. Figure 2(a) shows an example. Meanwhile, the up and bottom parts of image area are removed

due to the fact that they are normally the sky and the ground (Fig. 2(b)). This greatly reduces the time-consumed of dealing with the panorama and decreases the occurrence of false vanishing points detection.

**Line Segments Detection.** A general and fast approach by Bazin *et al.* [2] is adopted in our algorithm to detect line segments in the remained building segments. The retaining line segments meet two conditions, one is that their lengths exceed a certain threshold, the other is that they are above the horizon line, as shown in Fig. 2(b).



**Fig. 2.** Generating the VPs hypotheses. **(a)**: pixel-wise semantic segmentation of the panoramic image and selected building facade segment. **(b)**: detection of the line segments in building facade **(c)**: classification of the line segments using the estimated VPs. **(d)**: VPs detection and the VP-corresponding line clusters.

**Vanishing Points Estimation.** We use VP estimation approach [3] based on the RANSAC framework. This approach select the line segments to compute the VPs randomly at first. After RANSAC iteration, we get the best result of VPs which has the highest number of inliers. It is worth mentioning that there exit six VPs, all of which can be marked out in a single panorama, as illustrated in Fig. 2(c), (d).

Given the normal vector  $\mathbf{n}_{\mathbf{vp}_3}$  which is the vertical direction of  $\mathbf{vp}_3$ , we can calculate the roll  $\rho$  and pitch  $\varphi$  angle.

$$\begin{aligned} \mathbf{n}_{\mathbf{vp}_3} &= (\mathbf{n}_x, \mathbf{n}_y, \mathbf{n}_z) \\ \rho &= \arctan(\mathbf{n}_y / \mathbf{n}_z) \\ \varphi &= \arctan\left(\frac{-\mathbf{n}_z}{\sqrt{\mathbf{n}_y^2 + \mathbf{n}_x^2}}\right) \end{aligned} \quad (1)$$

**Estimating Orientation.** The absolute camera orientation (yaw) is defined as the angle between the direction of the omnidirectional camera and the due north. The included angle  $\Delta\theta$  is the relative angle between the horizontal vanishing point and horizontal axis.

$$direction_c = \theta + \Delta\theta \quad (2)$$

The horizontal direction is an additive combination of two parts, the rough camera orientation from compass and the included angle. Taking the compass information as a prior, we estimate the horizontal direction and thereby refine the camera orientation in the absolute map coordinate system.

### 2.3 Estimating Translation

In this step, we attempt to estimate the global 3D position of camera in an absolute coordinate system by seeking correspondences 2D building vertical edges in the image and the 3D building facade outlines in the map. In theory, the pose could be computed from alignments of the city-model maps with the panorama [1]. This procedure of estimating the position can be divided into four steps.

We start with extraction of building vertical corner edges from the panorama. In the second step, we perform the coordinate transformation between latitude, longitude and Universal Transverse Mercator (UTM). Next, we search the correspondences between the semantic segmentation panorama and the 2.5D map. As the last step, we detail the translation estimation.

We generate translation results for each possible pair of correspondences between the building vertical edges and the building outline. Given two normal vectors,  $\mathbf{n}_1$  and  $\mathbf{n}_2$  represent the two vertical edges in the image respectively, and two 3D points on building outlines,  $\mathbf{x}_1$  and  $\mathbf{x}_2$ , the translation  $\mathbf{t}$  in the ground plane can be easily computed by solving the following linear system:

$$\begin{aligned}\mathbf{n}_1 \cdot (\mathbf{x}_1 + \mathbf{t}) &= 0 \\ \mathbf{n}_2 \cdot (\mathbf{x}_2 + \mathbf{t}) &= 0\end{aligned}\tag{3}$$

We filter the results set based on their estimated 3D location. First, the translations which are located within the buildings are removed. Second, we define the circle that taking the camera as the center and whose radius is 12.5 m according to [1]. The results which are outside the circle are discarded. In addition, we set the height of the camera at 1.6 m. Due to the fact that we use the 2.5D city with global scale, the resulting translation has absolute scale.

## 3 Experiment

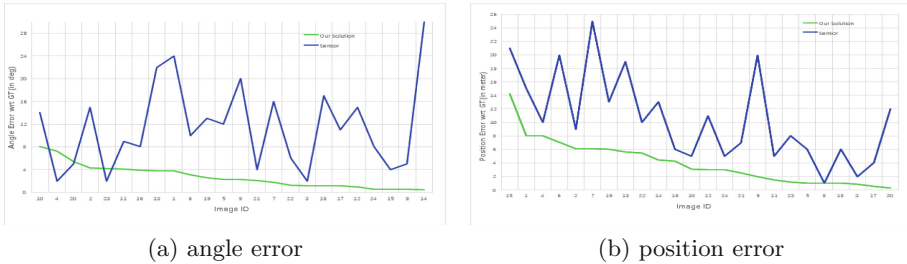
Due to the lack of public implementation of most similar studies, we cannot quantitatively compare with competing methods. We therefore use the rough pose information which comes from consumer sensors (e.g. GPS and IMU) as the comparison object. Moreover, because there is no standard public dataset consisting of accurate ground truth global position, panoramic images and corresponding 2.5D maps, we construct our own challenging dataset for validation.

### 3.1 Dataset

Our dataset comes from different places in the university in urban environment of Hangzhou. For validating our system, dataset includes different regional 2.5D maps, panoramic images. The 2.5D city map can easily obtained from broadly available map software, such as OpenStreetMap. In addition, the ground truth locations are recorded from the differential GPS equipment (with the precision of centimeter level). According to [1], we use the same technique to calculate ground truth pose. The technique consists of manually matching the 2D image locations with the 3D points from maps.

### 3.2 Pose Accuracy

**Orientation Accuracy.** The comparison line chart Fig. 3a directly reflects the angle error from consumer sensors (IMU) and our method. The angle error is defined as the discrepancy between the estimation rotation and the ground truth value. The images are ordered from the one with the largest angle error to the smallest. By adopting our method, the error is stable within  $8^\circ$ , and the vast majority of them is less than  $4^\circ$ . In addition, there may be a floating range of 1 to  $2^\circ$  in that way to calculating the ground truth pose due to the camera imaging model and the far distance of buildings from the camera.



**Fig. 3.** Pose error comparison. **(a):** Angle error comparison. **(b):** Position error comparison. The images are ordered from the one with the largest error to the smallest by two methods, which are from consumer sensors (GPS, IMU) and from our method respectively. We stay a significant lead both in angle accuracy and position accuracy.

**Translation Accuracy.** As for the position, we use the differential GPS equipment to record the ground truth location of image. We rank the image in the same way as the top line chat, as shown in Fig. 3b. Experiments demonstrate that 96% estimated translation from our method are more precise than the result from sensors (consumer GPS). The worst results may be caused by the poor segmentation results and the inaccurate 2.5D maps.

### 3.3 Time Cost

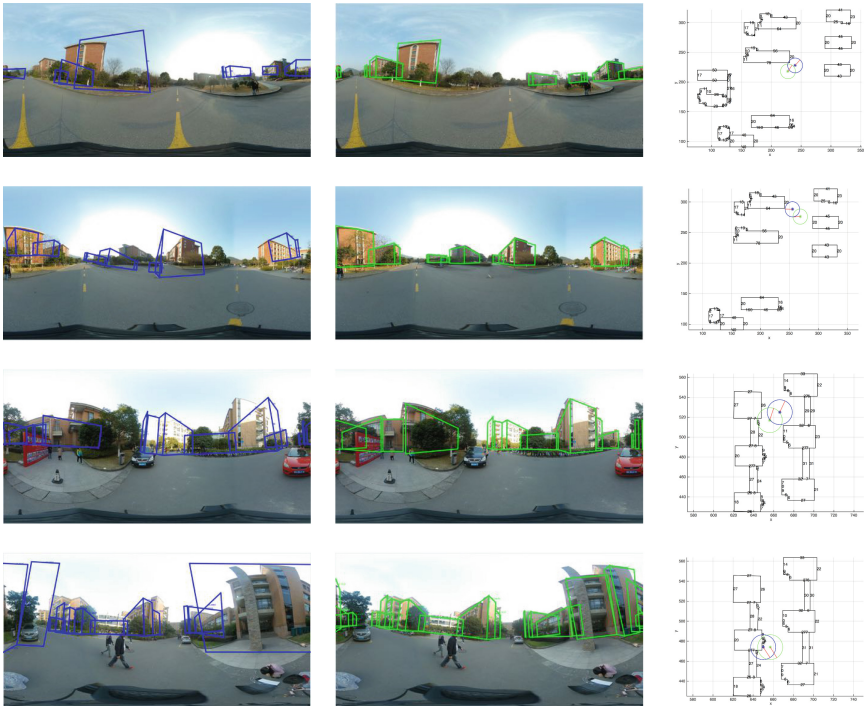
We test our method of pose estimation using Matlab code, and run it in a single CPU core on 2016 Inter 2 GHz i5 Macbook Pro for 2880\*1440 image. In the process of semantic segmentation of images, we use an Inter i7 7700k CPU, a Nvidia Gtx1080 GPU and 8 GB memory for each 1440\*720 image. The execution time for main parts of our code is shown on Table 1. The consuming-time mentioned in [1] is about 30(s) using a traditional camera. Although using the panoramic image, the time efficiency of our algorithm is increased by approximately 63% compared to [1].

**Table 1.** The timings of main parts of our algorithm.

Part	Algorithm	Approx. time(s)
1	Line segments detection	6.84
2	Semantic segmentation	0.33
3	Rotation estimation	0.90
4	Translation estimation	1.12
5	IO	2.21
6	<b>Total time consumed</b>	11.4

### 3.4 Visual Inspection

For each panoramic image, we reproject the 3D coordinate of buildings outline in map into the panoramic image. We chose the pose with better matching re-projection effect. By visual inspection, the difference of the estimated pose by our



**Fig. 4.** **Left:** Model reprojection into the panorama using the consumer sensor pose. **Middle:** Model reprojection into the panorama using our pose. **Right:** The difference of position between the consumer sensors (blue) and our method (green). The red line segments represent the camera orientation. (Color figure online)

method and the pose from rough sensors can be intuitively observed in Fig. 4. The model outlines can nicely fit the building in the image by our method.

## 4 Conclusion

In this paper, we present a method for 7-DOF pose estimation (orientation, translation, scale) based on the spherical panoramic imaging model using 2.5D maps. Our method therefore is evaluated on a challenging and real-scene dataset including full field-of-view panoramas, sensors data and maps. The results manifest the pose estimation in terms of rotation and translation enjoys an inspiring accuracy and a better running-time. Our method significantly improves the time-consumed by approximately 63% compared to the previous method.

**Acknowledgment.** This work was supported by the National Natural Science Foundation of China under Grant 61701442, the Natural Science Foundation of Zhejiang Province under Grant number Y18F030070.

## References

1. Arth, C., Pirchheim, C., Ventura, J., Schmalstieg, D., Lepetit, V.: Instant outdoor localization and slam initialization from 2.5D maps. *IEEE Trans. Vis. Comput. Graph.* **21**(11), 1309–1318 (2015)
2. Bazin, J.C., Demonceaux, C., Vasseur, P., Kweon, I.: Rotation estimation and vanishing point extraction by omnidirectional vision in urban environment. *Int. J. Robot. Res.* **31**(1), 63–81 (2012)
3. Bazin, J.C., Pollefeys, M.: 3-line ransac for orthogonal vanishing point detection. In: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 4282–4287. IEEE (2012)
4. Chu, H., Gallagher, A., Chen, T.: GPS refinement and camera orientation estimation from a single image and a 2D map. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 171–178 (2014)
5. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431–3440 (2015)
6. Mur-Artal, R., Montiel, J.M.M., Tardos, J.D.: ORB-SLAM: a versatile and accurate monocular slam system. *IEEE Trans. Robot.* **31**(5), 1147–1163 (2015)
7. Schindler, G., Brown, M., Szeliski, R.: City-scale location recognition. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2007, pp. 1–7. IEEE (2007)
8. Taneja, A., Ballan, L., Pollefeys, M.: Registration of spherical panoramic images with cadastral 3D models. In: 2012 Second International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT), pp. 479–486. IEEE (2012)
9. Ventura, J., Höllerer, T.: Wide-area scene mapping for mobile visual tracking. In: 2012 IEEE International Symposium on Mixed and Augmented Reality (ISMAR), pp. 3–12. IEEE (2012)
10. Zandbergen, P.A., Barbeau, S.J.: Positional accuracy of assisted GPS data from high-sensitivity GPS-enabled mobile phones. *J. Navig.* **64**(3), 381–399 (2011)





# An Adaptive Clustering Algorithm by Finding Density Peaks

Juanying Xie<sup>(✉)</sup>  and Weiliang Jiang

School of Computer Science, Shaanxi Normal University,  
Xi'an 710062, People's Republic of China  
xiejuany@snnu.edu.cn

**Abstract.** Clustering by fast search and find of density peaks (shorted as DPC) is a powerful clustering algorithm. However it has a fatal problem that once a point is assigned erroneously, then there may be many more points will be assigned to error clusters. Furthermore, its density peaks need to be selected manually, so as to the clustering may be poor. Lastly it cannot find density peaks from sparse cluster when the data set comprises dense and sparse clusters simultaneously. This paper proposed a new clustering algorithm to overcome the aforementioned weaknesses of DPC by adaptively finding density peaks and assigning points to their most proper clusters. The new density  $\rho_i$  of point  $i$  was defined. The adjusting strategy for  $\gamma_i$ , and the assignment strategy for remaining points, and the merging strategy for erroneously partitioned clusters were proposed. Many challengeable synthetic datasets were used to test the power of the proposed algorithm. The experimental results demonstrate that the proposed algorithm can correctly detect clusters with any arbitrary shapes. Its performance is superior to DPC and its variants in terms of bench mark metrics, such as clustering accuracy (Acc), adjusted mutual information (AMI) and adjusted rand index (ARI).

**Keywords:** Density peaks · Standard deviation · Clustering

## 1 Introduction

Clustering is to group similar points into same clusters and dissimilar ones into other clusters. It is an unsupervised learning process. Its motivation is to find the patterns, rules and knowledge embedded in the data set. It has become more and more popular with the increasing aggregate data in the real world.

There are many different kinds of clustering algorithms, such as K-means [6], K-medoids [5] and their variants [10]. These clustering algorithms are based on partition ideas, and cannot detect non-spherical clusters in a data set. The density based clustering algorithms, such as DBSCAN (Density-Based Spatial Clustering of Application with Noise) [7] etc., can detect clusters with any arbitrary

---

Supported by NSFC under Grant No. 61673251 & by Fundamental Research Funds for Central Universities under Grant No. GK201701006.

shapes and avoid the limitations from partitioning based clustering algorithms. However, DBSCAN is a time consuming algorithm and cannot be used to detect clusters in big data. Although some other famous clustering algorithms have been proposed [2], time consuming load is still exist. Clustering by fast search and find of density peaks was proposed in 2014 by Rodríguez and Laio [11]. We name it as DPC (Density Peaks finding Clustering) for short. It is a very fast clustering algorithm. Its most contribution is that it coined local density and distance for a point, and proposed the decision graph with density and distance as x-axis and y-axis respectively, so as to scatter all points in the decision graph and select upright corner points as density peaks, that is cluster centers. After that each remaining point was assigned to its nearest neighbor with higher density. Its power was tested in [11]. Although it is a very powerful clustering algorithm, it has got several limitations. First it uses different local density definition for small and big data respectively, but the borderline between small and big data is vague. Second its assignment strategy may lead to the “domino effect”, that is an error propagation, where once a point is assigned to an error cluster, then there may be many more points will be assigned erroneously, so as to a poor clustering will be produced. It’s third limitation is that it determine the density peaks manually. Fourth its density definition may make the density peaks cannot be detected from sparse clusters. Therefor several variants of DPC have been proposed [8,14–16] to remedy the aforementioned limitations in it. This paper will advance DPC by adaptively detecting density peaks and finding the potential pattern in a data set.

The details of our proposed algorithm will be introduced in Sect. 2. Section 3 will display experiments and analyses. Conclusions will come in Sect. 4.

## 2 Proposed Algorithm

In order to detect density peaks of a data set adaptively and avoid the error propagation of DPC, so as to find the genuine pattern of a data set, we propose a new method to detect density peaks adaptively, and a new assignment strategy to assign remaining points except for density peaks to their most proper clusters, and a merging strategy to merge clusters which should be the same one. Here are the innovations of our algorithm.

### 2.1 Detecting Density Peaks Adaptively

We define the local density  $\rho_i$  of point  $i$  in (1), where  $d_{ij}$  is the Euclidean distance between points  $i$  and  $j$ , and  $KNN_i$  is the  $K$  nearest neighbors of point  $i$ . Then we define the distance  $\delta_i$  and the  $\gamma_i$  ( $= \rho_i \times \delta_i$ ) of point  $i$  as that in [11]. After that we rank points in their  $\gamma$  values in descending order to find the first point  $i$  that satisfies  $\gamma_i - \gamma_{i+1} > \bar{\gamma}$ , where  $\bar{\gamma} = \frac{\gamma_1 - \gamma_{100}}{100}$ . Consequently there are  $i$  density peaks to be found, that is, there are  $i$  cluster centers to be detected. It should be noted that the ranked  $\gamma$  in descending order guarantees that  $\bar{\gamma}$  is approximate the global one.

$$\rho_i = \frac{1}{\sqrt{\frac{1}{K-1} \sum_{j \in KNN_i} d_{ij}^2}} \tag{1}$$

**2.2 Adjusting Strategy**

DPC cannot find cluster centers from sparse clusters when a data set simultaneously comprises dense and sparse clusters [8,16]. To overcome the weakness, we proposed to reduce the weight of local density  $\rho_i$  of point  $i$  while emphasizing the effect of its distance  $\delta_i$  on its  $\gamma_i$  by  $\gamma_i = \rho_i^{0.1} \times \delta_i$ , so as to detect all density peaks no matter they come from dense or sparse clusters.

**2.3 Assignment Strategy**

To overcome the limitations from the assignment strategy of DPC, we proposed the new KNN (K-Nearest Neighbors) based assignment strategy to assign the remaining points after cluster centers have been found. Our idea is to assign K nearest neighbors of a cluster center to it; then for a remaining point  $i$ , we first find its K nearest neighbors with higher densities and having been assigned, then find its K nearest neighbors having been assigned no matter how their densities are. We assign point  $i$  to the same cluster as majority points in the intersection of the two kinds of K nearest neighbors of point  $i$ . If the intersection is empty, then the point  $i$  will be assigned to the same cluster as its nearest neighbor with higher density as DPC does.

**2.4 Merging Strategy**

Sometimes there are more density peaks having been found, so as to there are more clusters being detected. Therefore, when we get a clustering, we evaluate whether clusters of points  $i$  and  $j$  should be merged or not. If the distance  $d_{ij}$  between points  $i$  and  $j$  from different clusters is less than the average of their local standard deviation, that is  $d_{ij} < \frac{dc(i)+dc(j)}{2}$ , where  $dc(i) = \sqrt{\frac{1}{K-1} \sum_{j \in KNN_i} d_{ij}^2}$ , and  $\rho_i$  and  $\rho_j$  satisfy the Eq.(2), where  $\alpha$  is a threshold around 1 and  $\bar{\rho}(cl(i))$  is the average density of points from cluster of point  $i$ .

$$\rho_i > \alpha \times \bar{\rho}(cl(i)) \parallel \rho_j > \alpha \times \bar{\rho}(cl(j)) \tag{2}$$

**2.5 Main Steps of Proposed Algorithm**

The specific steps of proposed algorithm are in Algorithm 1.

**Input:** data set  $\mathbf{X} = \{\mathbf{x}_i, i = 1, \dots, n\}$ , the parameter  $K$ , and the threshold  $\alpha$ .

**Output:** a clustering of  $\mathbf{X}$ .

normalize data with min-max normalization;

calculate the local density  $\rho_i$  of point  $i$  in (1);

calculate the distance  $\delta_i$  of point  $i$ ;

**if** the data set is balanced **then**

```

|   for  $i = 1$  to  $n$  do
|   |    $\gamma_i = \rho_i \times \delta_i$ ;
|   end

```

**end**

**else**

```

|   for  $i = 1$  to  $n$  do
|   |    $\gamma_i = \rho_i^{0.1} \times \delta_i$ ;
|   end

```

**end**

rank points in descending order by their  $\gamma$  values;

select density peaks adaptively in the way described in 2.1;

assign remaining points by assignment strategy described in 2.3;

merge clusters that should be the same one by merging strategy described in 2.4;

output clustering finally detected;

**Algorithm 1.** The proposed adaptive clustering algorithm

### 3 Experiments and Analyses

We test the power of proposed algorithm, and compare its performance to that of DPC [11], KNN-DPC [14], FKNN-DPC [15], SD\_DPC [16], and MehmoodDPC [8] in terms of Acc, AMI and ARI. These three metrics are very famous benchmark metrics to test the performance of a clustering algorithm [9]. We have done the statistic test to reveal the significant differences between 6 clustering algorithms, but we cannot include the detail results of statistic test in this paper for the limit in pages. To express conveniently, we name the proposed algorithm as ADA-DPC.

We do not compare the performance of ADA-DPC to that of typical density based clustering algorithm DBSCAN, and the other very famous clustering algorithms because it has been demonstrated that DPC, KNN-DPC and FKNN-DPC are superior to DBSCAN and other famous clustering algorithms in [14, 15].

#### 3.1 Descriptions to Datasets

We did experiments on typical datasets from references [1, 3, 4, 12, 13, 17, 18] and on synthetic ones to test the power of proposed ADA-DPC. We did experiments on 22 datasets, but because of the limit in pages, we cannot display all results. Table 1 display datasets on which the experimental results will be shown in this paper. Table 2 displays the parameters to generate *dataset3*. Tables 3 and 4 show the parameters to generate datasets of *disk* and *ellipse*, respectively. Dataset *spiralsquare* is generated by adding two square shape clusters to the data set from references, so we do not give parameters to generate it.

**Table 1.** Datasets used in experiments.

Data set	# points	# attributes	# clusters
a3	7500	2	50
target	770	2	6
jain	373	2	2
compound	399	2	6
zelnik1	299	2	3
chainlink	1000	3	2
spiralsquare	13500	2	8
complex9	3031	2	9
ellipse	8002	2	2
disk	8106	2	2
dataset3	20000	2	4

**Table 2.** Parameters to generate *dataset3*.

Parameters	Cluster1	Cluster2	Cluster3	Cluster4
Mean	[2, 2]	[9, 2]	[6, 5.5]	$x \in [0.5, 12.5]$
Covariance	$\begin{bmatrix} 0.2, & 0 \\ 0, & 0.2 \end{bmatrix}$	$\begin{bmatrix} 1, & 0 \\ 0, & 1 \end{bmatrix}$	$\begin{bmatrix} 1.5, & 0 \\ 0, & 1.5 \end{bmatrix}$	$y \in [8, 9.5]$
# points	8,000	3,000	3,999	5,001

**Table 3.** Parameters to generate *disk*.

Parameters	Cluster1	Cluster2
Mean	$x \in [2, 2]$	[0, 0]
Covariance	$y \in ((\sqrt{1-x^2}, \sqrt{4-x^2}) \cup (-\sqrt{4-x^2}, -\sqrt{1-x^2}))$	$\begin{bmatrix} 0.05, & 0 \\ 0, & 0.05 \end{bmatrix}$
# points	8,006	100

**Table 4.** parameters to generate *ellipse*.

Parameters	Cluster1	Cluster2
Mean	$t \in [0, \pi], x = \cos(t)$	[0, 0]
Covariance	$y \in ((2 \times \sqrt{1-x^2}, 3 \times \sqrt{1-x^2}) \cup (-3 \times \sqrt{1-x^2}, -2 \times \sqrt{1-x^2}))$	$\begin{bmatrix} 0.1, & 0 \\ 0, & 0.1 \end{bmatrix}$
# points	4,002	4000

### 3.2 Experimental Results and Analyses

Here are the performance comparison between our ADA-DPC and DPC, KNN-DPC, FKNN-DPC, SD\_DPC, and MehmoodDPC. Figures 1, 2 and 3 are the clustering results in terms of Acc, AMI and ARI respectively.

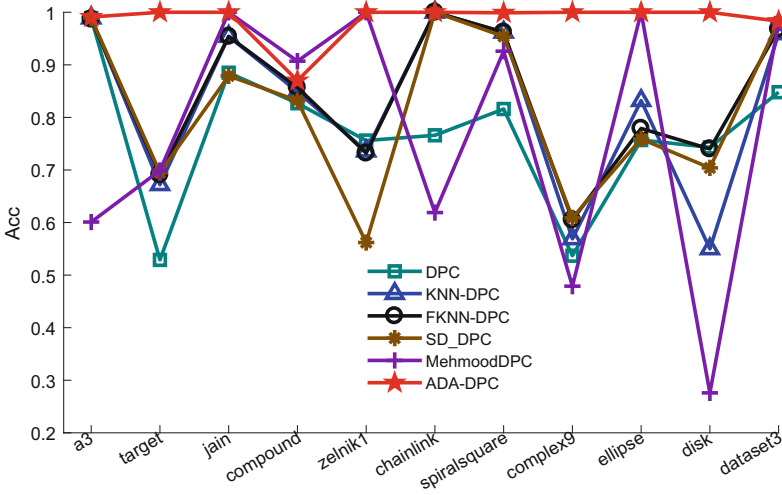


Fig. 1. The Acc comparison of 6 clustering algorithms on datasets in Table 1.

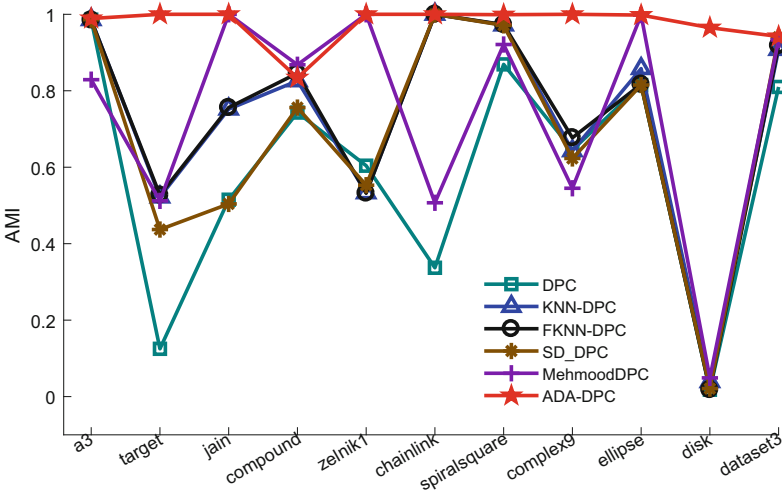
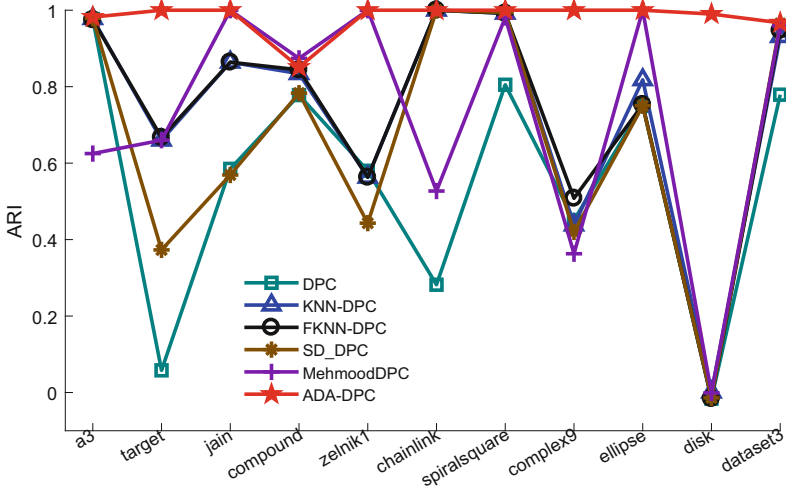


Fig. 2. The AMI comparison of 6 clustering algorithms on datasets in Table 1.



**Fig. 3.** The ARI comparison of 6 clustering algorithms on datasets in Table 1.

From the quantity comparison of clustering results of 6 clustering algorithms in terms of Acc, AMI and ARI respectively shown in Figs. 1, 2 and 3, we can see that our ADA-DPC is superior to other 5 clustering algorithms. Its performance definitely whelms that of DPC, KNN-DPC, FKNN-DPC, and SD\_DPC except for less being defeated by MehmoodDPC on *compound* data set.

We further studied the *compound* data set from [17], and found it is a very challengeable data set comprising dense and sparse clusters simultaneously while with any arbitrary shape clusters. Our ADA-DPC can nearly recognize all clusters in *compound* except for its failing in detecting swallow shape cluster from its background sparse cluster. Although MehmoodDPC can detect the correct number of clusters of *compound*, it cannot completely detect the sparse cluster around swallow shape. We did not display the clusterings of *compound* by 6 clustering algorithms for the limit in pages. In addition, although the pages are limit, we cannot bury the statistic test results that our ADA-DPC is significantly different from other 5 clustering algorithms.

All of the results demonstrate that the proposed ADA-DPC can recognize the clusters with any arbitrary shapes, and can find the genuine clustering of a data set. It outperforms the compared clustering algorithms of DPC, KNN-DPC, FKNN-DPC, SD\_DPC, and MehmoodDPC.

## 4 Conclusions

An adaptive clustering algorithm named ADA-DPC was proposed in this paper. It can adaptively find the cluster centers and detect the genuine clustering of a data set. The experiments on many challengeable datasets demonstrate that

the proposed ADA-DPC outperforms the available same kind of clustering algorithms. It has provided the way to solve the challengeable problems in data science to find the genuine pattern in a data set.

## References

1. Fränti, P., Virtajoki, O.: Iterative shrinking method for clustering problems. *Pattern Recogn.* **39**(5), 761–775 (2006)
2. Frey, B.J., Dueck, D.: Clustering by passing messages between data points. *Science* **315**(5814), 972–976 (2007)
3. Jain, A.K., Law, M.H.C.: Data clustering: a user’s dilemma. In: Pal, S.K., Bandyopadhyay, S., Biswas, S. (eds.) *PREMI 2005*. LNCS, vol. 3776, pp. 1–10. Springer, Heidelberg (2005). [https://doi.org/10.1007/11590316\\_1](https://doi.org/10.1007/11590316_1)
4. Karkkainen, I., Franti, P.: Dynamic local search for clustering with unknown number of clusters. In: *Proceedings of the 16th International Conference on Pattern Recognition*, vol. 2, pp. 240–243. IEEE (2002)
5. Kaufmann, L., Rousseeuw, P.J.: Clustering by means of medoids. In: Dodge, Y. (ed.) *Statistical Data Analysis Based on the L1-Norm and Related Methods*, pp. 405–416. North-Holland, Amsterdam (1987)
6. 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*, vol. 1: Statistics, Oakland, CA, USA, pp. 281–297 (1967)
7. Martin Ester, Hans-Peter Kriegel, J.S., Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD 1996)*, Portland, Oregon, USA, pp. 226–231 (1996)
8. Mehmood, R., EI-ASHram, S., Bie, R., Dawood, H., Kos, A.: Clustering by fast search and merge local density peaks for gene expression microarray data. *Sci. Rep.* **7**, 45602 (2017)
9. Nguyen, X.V., Epps, J., Bailey, J.: Information theoretic measures for clusterings comparison: is a correction for chance necessary? In: *Proceedings of the 26th Annual International Conference on Machine Learning, ICML 2009*, Montreal, Quebec, Canada, 14–18 June 2009, pp. 1073–1080 (2009)
10. Park, H.S., Jun, C.H.: A simple and fast algorithm for k-medoids clustering. *Expert Syst. Appl.* **36**(2), 3336–3341 (2009)
11. Rodríguez, A., Laio, A.: Clustering by fast search and find of density peaks. *Science* **344**(6191), 1492–1496 (2014)
12. Salvador, S., Chan, P.: Determining the number of clusters/segments in hierarchical clustering/segmentation algorithms. In: *IEEE International Conference on TOOLS with Artificial Intelligence*, pp. 576–584 (2004)
13. Ultsch, A.: Clustering with SOM: U\*C. In: *Proceedings of WSOM 2005*, pp. 336–337 (2005)
14. Xie, J., Gao, H., Xie, W.: K-nearest neighbors optimized clustering algorithm by fast search and finding the density peaks of a dataset. *SCIENTIA SINICA Informationis* **46**(2), 258–280 (2016)
15. Xie, J., Gao, H., Xie, W., Liu, X., Grant, P.W.: Robust clustering by detecting density peaks and assigning points based on fuzzy weighted k-nearest neighbors. *Inf. Sci.* **354**, 19–40 (2016)



16. Xie, J., Jiang, W., Ding, L.: Clustering by searching density peaks via local standard deviation. In: Yin, H., et al. (eds.) IDEAL 2017. LNCS, vol. 10585, pp. 295–305. Springer, Cham (2017). [https://doi.org/10.1007/978-3-319-68935-7\\_33](https://doi.org/10.1007/978-3-319-68935-7_33)
17. Zahn, C.T.: Graph-theoretical methods for detecting and describing gestalt clusters. *IEEE Trans. Comput.* **C-20**(1), 68–86 (2006)
18. Zelnik-Manor, L.: Self-tuning spectral clustering. In: *Advances in Neural Information Processing Systems*, vol. 17, pp. 1601–1608 (2004)



# Statutes Recommendation Using Classification and Co-occurrence Between Statutes

Yi Feng, Jidong Ge<sup>(✉)</sup>, Chuanyi Li<sup>(✉)</sup>, Li Kong,  
Feifei Zhang, and Bin Luo

State Key Laboratory for Novel Software Technology, Software Institute,  
Nanjing University, Nanjing 210093, China  
gjdnu@163.com, lcynju@126.com

**Abstract.** In the trial process, it is difficult and tedious for judges to find appropriate statutes to decide cases, especially complicated cases. In this paper, we propose a method to recommend statutes that are applicable to judging new cases for judges. Our method utilizes the associations between causes of action and statutes as well as the co-occurrence among statutes to predict applicable statutes based on Artificial Neural Networks. The experiment data are all from the real court judgments. Our experimental results show that our method can effectively and accurately recommend statutes that are more likely to appear in real judgments. The proposed method gets better results compared to several baselines.

**Keywords:** Statutes recommendation · Multi-label classification

## 1 Introduction

Artificial intelligence and law has attracted much attention. Many scholars have made a lot of important research work. The application of artificial intelligence in the legal domain can be divided into four categories, which are legal search, document review, case prediction and consulting services. In legal domain, there are tens of thousands of statutes. When deciding new cases, it is difficult for judges to find appropriate statutes as legal basis, especially complicated cases. In this paper, we study the problem of statutes recommendation. With statutes recommendation, we can provide judges with applicable statutes easily, which undoubtedly improve efficiency and reduce their workload.

On the basis of analyzing a large number of judgment documents, we not only find out the association between causes of action (CoA) and statutes but also the co-occurrence between statutes. We propose a recommendation method using classification and co-occurrence. Given detail description of cases, our method predicts CoA for each case by the CoA classifier to identify associated statutes, which reduces classification categories. Then the statutes classifier correspond to the CoA is used to recommend statutes. Based on the probability of the recommended results, the top  $k_1$  statutes are selected. Then, we resort the top  $k_1$  statutes based on co-occurrence between them. At last, we take the top  $k_2$  statutes as the final results. Compared with several baseline, our method has better results.

## 2 Related Work

In general, recommendation algorithms fall into two main categories, which are content-based [1, 2] and collaborative filtering [3]. Among them, collaborative filtering is the mainstream method. If further divided, collaborative filtering can be divided into memory-based [4] and model-based [5]. In recent years, some recommended algorithms based on neural networks have been proposed. Yu et al. [6] used neural networks to learn the vector representations for both users and microblog texts to recommend microblogs. Wang et al. [7] used a single neural network to model users and products, generating customized product representations using a deep memory network, from which customized ratings and reviews are constructed jointly. In the legal domain, Liu et al. [8] used the classification algorithm to deal with statutes recommendation. They use multi-label Support Vector Machine (SVM) [9] to select the top  $k$  statutes and get the most similar statutes according to the statutes' text. However, the number of statutes is large, and it leads to too many categories. The method proposed in our paper also converts statutes recommendation into a classification problem. We predict CoA for each case, which reduces categories significantly.

## 3 Approach

### 3.1 Legal Background

After judgments, all proceedings of trials are recorded in documents, called judgment documents. A judgment document records the whole process of a trial, such as evidence, facts and cited statutes. In this paper, we utilize judgment documents as our training data. The CoA is a brief summary of case contents. For example, divorce cases are classified as divorce dispute. Traffic accident cases belong to traffic dispute. CoAs are associated with statutes. Some statutes are more applicable to some CoAs. For example, criminal law is used to judge criminal cases. It is impossible to use marriage law to sentence murders. When recommending statutes, we can determine the CoA and put more attention to statutes associated with this CoA.

### 3.2 Overview

In this paper, we take advantage of classification technologies to recommend statutes. Statutes are seen as categories. There are some unique differences that need to be addressed. It is difficult to recommend so many statutes, which results in too many categories. A judgment document may cite one or some statutes. It is a multi-label classification problem. Besides these, statutes may be cited together. There are co-occurrences between statutes, i.e., categories are not independent.

In view of the above characteristics, we propose a novel method to recommend statutes. Figure 1 shows the overview of our proposed approach. Given detail description of cases, we preprocess them, including segment, stop-words removal, characterization and dimension reduction. We utilize *TF-IDF* to represent judgment documents and Singular Value Decomposition (SVD) is taken to reduce dimension.

In order to solve the problem of excessive categories, we first predict the CoA to narrow the scope of applicable statutes and put more attention to particular statutes that are associated with the CoA. After determine the CoA, statutes classifiers are taken to predicting applicable statutes. The top  $k_1$  statutes which have high probability are taken into account. Some statutes are more likely to appear together than others. In order to get more precise recommended sequence, we utilize co-occurrence relationships between these top  $k_1$  statutes to resort them. Statutes that have co-occurrence relationships should be recommended together than other statutes. After resorting, the top  $k_2$  ( $k_2 < k_1$ ) in the top  $k_1$  are selected as the final results.

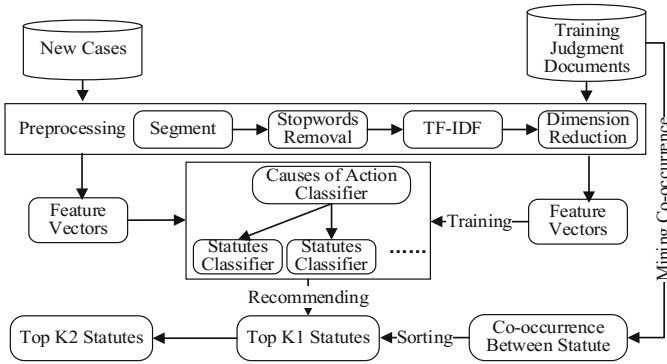


Fig. 1. The overview of our proposed approach

### 3.3 Predicting CoAs

In this step, we predict CoA for each judgment document. Artificial Neural Network (ANN) is applied to the training collection of judgment documents to generate a CoA classifier. The inputs of the CoA classifier are  $TF - IDF$  vectors. The outputs are corresponding CoAs. Let  $(x^{(i)}, y^{(i)})$  denotes the  $i$ -th group of the training set, where  $x^{(i)}$  represents the  $TF - IDF$  vector and  $y^{(i)}$  is the  $k$ -dimensional CoA vector. There are  $m$  samples. The training object is to minimize the cross entropy cost:

$$J_{\theta} = -\frac{1}{m} \sum_{i=1}^m \sum_1^k y_k^{(i)} \log(h_{\theta}(x^{(i)}))_k + (1 - y_k^{(i)}) \log(1 - h_{\theta}(x^{(i)}))_k + \frac{\lambda}{2m} \sum_w w^2 \quad (1)$$

Where  $h_{\theta}(x^{(i)})$  denotes the ANN predictive value vector and  $\theta$  are parameters of the CoA classifier. Overfitting is a major problem in neural networks. Our CoA classifier has very large numbers of weights and biases. Thus we use  $L_2$  regularization technique to reduce overfitting. We add an extra term to the cost function. We choose Stochastic Gradient Descent (SGD) algorithm to train our network. In addition, we take some techniques to make weights learn fast and avoid a learning slow down. Suppose we have a neuron  $n_{in}$  input weights. Then we initialize those weights as Gaussian random variables with mean 0 and standard deviation  $1/\sqrt{n_{in}}$ . We continue choose the bias as a Gaussian with mean 0 and standard deviation 1.

### 3.4 Recommending Statutes

After predicting the CoA, we recommend statutes related this CoA and ignore these unassociated ones. In this step, we still take advantage of ANN to recommend statutes. The inputs of ANNs are *TF - IDF* vectors too. The outputs are applicable statutes. We train different statutes ANNs for each CoA. The output dimension is the number of statutes related to the CoA. If there are  $n$  statutes associated with one CoA, the output dimension is  $n$ . The training object is same as the CoA classifier. We apply a softmax layer in the last to make the summation of all the outputs equal to 1. Each neuron of the last layer output the applicable probability of the statute. We take the top  $k_1$  statutes to recommend based on the probability.

### 3.5 Resorting Statutes

Some statutes are often cited together. We take co-occurrence relationships into account to get a more accurate recommendation results. In this paper, we propose a ranking method to resort the recommended statutes to obtain more accurate results. The method is divided into two steps: (1) Mining association rules between statutes; (2) Resorting based on association rules. We can use follow two association rules to represent the relationship between two statutes:

$$Support(S_1 \rightarrow S_2) = freq(S_1, S_2); Confidence(S_1 \rightarrow S_2) = \frac{freq(S_1, S_2)}{freq(S_1)} \quad (1)$$

Where  $freq(S_1, S_2)$  represents the frequency of  $S_1$  and  $S_2$  being cited together.  $freq(S_1)$  represents the number of occurrences of  $S_1$ .  $Support(S_1 \rightarrow S_2)$  means the probability that  $S_1$  and  $S_2$  appear at the same time.  $Confidence(S_1 \rightarrow S_2)$  means the probability of citing  $S_2$  while citing  $S_1$ . For example,  $S_1 \rightarrow S_2[Support = 10\%; Confidence = 60\%]$ . It means  $S_1, S_2$  appear together in 10% of the judgment documents. And 60% of the documents citing  $S_1$  also refer to  $S_2$ . If  $S_1$  is cited,  $S_2$  has a 60% chance to be cited too. We use a voting algorithm inspired by TextRank to calculate the impact of association rules on the final recommended probability of each statute. The algorithm can be expressed as a directed graph  $G = (V, E)$ , which is composed of a set of points  $V$  and edges  $E$ . In the graph, the weight of the edge from  $v_i$  to  $v_j$  is  $w_{ij}$ . For a given point  $v_i$ ,  $In(v_i)$  is the point set pointing to  $v_i$ , and  $Out(v_i)$  is the point set pointed by point  $v_i$ . The score of point  $v_i$  is defined as follows:

$$WS(v_i) = (1 - d) + d * \sum_{v_j \in In(v_i)} \frac{w_{ji}}{\sum_{v_k \in Out(v_j)} w_{jk}} WS(v_j) \quad (2)$$

Where the points represent statutes. The weight of the edge is the *Confidence* between two statutes. Because the *Confidence* is directional, it is a directed graph. For example, if  $Confidence(S_1 \rightarrow S_2) = 60\%$ , there is a directional edge that points from  $S_1$  to  $S_2$ , and the weight is 60%. In the graph, each point votes for the points it points to. So if the value of a point is large, the points that pointed by this point are large too.

It means if a statute has a large citation probability, statutes that have association rules with it are more likely to be cited together. In the formula,  $d$  is a damping coefficient with a range of 0 to 1. At initialization, the value of each point  $v_i$  is 0. Then we loop the above formula until convergence. The final value of each point represents the impact of association rules. In the last, the final recommended probability of statute  $S_i$  is calculated by:

$$FP_{S_i} = P_i + WS(v_i) * \log(M + 1) \quad (3)$$

Where  $P_i$  is the recommended probability of  $S_i$  from the statutes classifier and  $M$  denotes the number of statutes that have association rules with  $S_i$  in the top  $k_1$  statutes. We resort the top  $k_1$  statutes based on the  $FP_{S_i}$  of each statute. After resorting, we recommend the top  $k_2$  ( $k_2 < k_1$ ) statutes as the final results.

## 4 Experiment

### 4.1 Data Collection

In order to test the effectiveness of our method in this paper, we downloaded about 70 thousand documents from China Judgment Online<sup>1</sup> to construct our dataset. This dataset contains six CoAs. The distribution of documents is shown in Table 1. CoAs are represented as numbers. Then we get detail description of cases, the CoAs, and the cited statutes from them.

**Table 1.** Dataset distribution

Causes of action	Number of documents	Training set	Test set
0	12755	11479	1276
1	11768	10591	1177
2	12548	11293	1255
3	10207	9186	1021
4	11130	10017	1113
5	12241	11016	1225
Total	70649	63582	7067

### 4.2 Architecture Details

In the preprocessing, we utilize Jieba<sup>2</sup> to segment words. The *TF - IDF* vectors are reduced to 1000 dimensions. The CoA classifier has five layers in total and another softmax layer. The output layer has six neurons corresponding to the six CoAs. The input layer has 1000 neurons. The three hidden layers are all 500-dimensional.

<sup>1</sup> <http://wenshu.court.gov.cn/>.

<sup>2</sup> <https://github.com/fxsjy/jieba>.

The activation functions are all sigmoid. The learning rate in the SGD is  $\eta = 0.1$  and the regularization parameter is  $\lambda = 0.1$ . We traverse all the data 40 times in the training process, i.e., epoch = 40. Each SGD update direction is computed with a mini batch of 100 samples. Each statutes classifier has 5 layers and another softmax layer. The input layer has 1000 neurons. The dimension of the output layer equals to the number of statutes associated with the corresponding CoA. The hidden layers are all 500-dimensional. Activation function is sigmoid. The learning rate in the SGD is  $\eta = 0.2$  and the regularization parameter is  $\lambda = 0.1$ . When training, we choose epoch as 40 and the minibatch is 50. The parameter  $d$  in the formula (3) is 0.85.

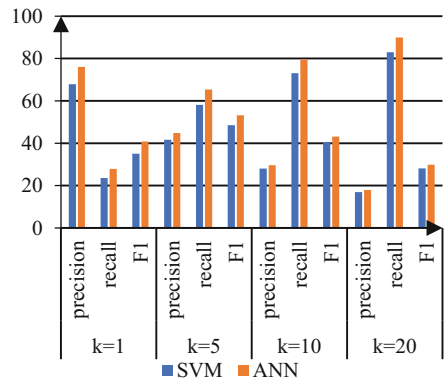
### 4.3 Experiments and Results

We utilize three metrics to evaluate:  $precision = \frac{1}{n} \sum \frac{k_r}{k}$ ,  $recall = \frac{1}{n} \sum \frac{k_r}{m}$ ,  $F1 = \frac{2 * precision * recall}{precision + recall}$ , where  $n$  is the total number of documents,  $k_r$  is the number of statutes which are predicted right in the  $k$  statutes,  $m$  is the number of statutes which is cited in fact in a document.

We compare ANN with other classification algorithms in predicting CoAs. We utilize two baselines, i.e., Decision Tree (DT) and SVM. We also compare ANN with SVM in predicting statutes. We take the top  $k$  statutes as the prediction results. The value of  $k$  is 1, 5, 10, and 20 respectively. The results are shown in Table 2 and Fig. 2.

**Table 2.** Results of predicting causes of action by different classifiers.

Classifier	Accuracy
DT	93.08%
SVM	87.93%
ANN	98.57%



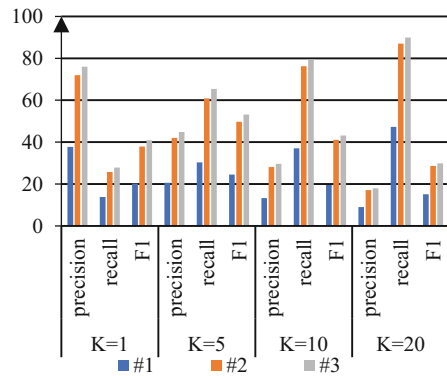
**Fig. 2.** Results of predicting statutes.

From results, we can see that ANN classifier is better than DT and SVM. ANN performs better than SVM in predicting statutes. With  $k$  increases, precision is lower and recall is higher. As  $k$  increases, the denominator of the precision formula increases. On the contrary,  $k_r$  in recall formula gets larger, resulting in higher recall. The average number of cited statutes is around 4. So  $k_r$  in the formula of precision and recall has a growth limit. It is why precision is so low when  $k$  is 20. When  $k$  is 1,  $k_r$  is 1 or 0, but  $m$  in recall formula is large, resulting in lowest recall.

Besides above, we use three models to verify whether the associations between CoAs and statutes can improve the effectiveness of statutes recommendation. The first model does not consider CoA and recommend statutes directly by SVM classifier, represented as #1. The second model #2 is same as the first model, but it utilizes ANN as classifier. The third model #3 is implemented based on our proposed approach and considers CoA, but does not take co-occurrence into account. The results are shown in Fig. 3. It is observed that the recommendation based on CoAs is better than direct recommendation. CoAs and statutes have some associations. A document with specific CoA always cites in a large number of specific statutes. And statutes are always cited by some specific CoAs. Therefore, we use associations to focus on particular statutes associated with CoA. It not only reduces recommendation categories, but also improve performance.

**Table 3.** F1-measure (%) comparison of the two models

k1	k2	#4	#5
5	1	40.8	41.3
10	5	53.2	53.6
20	10	43.1	43.4
30	20	29.9	30.1



**Fig. 3.** Comparison of the three models

In order to verify whether the co-occurrence can further improve the effectiveness of statutes recommendation, we use two models for comparison. The first model denoted as #4 is implemented according to our approach without co-occurrence analysis. The second model denoted as #5 adds co-occurrence analysis based on the first model. The results are shown in Table 3. The use of co-occurrence improves the final results. Some statutes are often cited together and more likely to appear together

We also compare our method with other baselines. We utilize topic models(LDA) to get the most similar documents and recommend cited statutes in them. Besides, we add a modified topic model Labeled LDA (LLDA) to enhance our comparison. The results are shown in Table 4. Our proposed model gets the best performance compared to other baselines.

Overall, the experimental results are encouraging. It shows that the two characteristics of judgment documents we find, i.e., the associations between CoAs and statutes, and the co-occurrence between statutes contribute to statutes recommendation. Compared with SVM, ANN does better in the judgment documents. Our method performs better than several baselines.



**Table 4.** Recall (%) compared with other models

K1	K2	TFIDF+SVM	TFIDF+Cosine	LDA	LLDA	Our method
5	1	13.8	16.1	14.2	15.6	28.2
10	5	30.3	29.8	27.0	37.6	65.4
20	10	37.0	50.5	43.1	49.7	79.5
30	20	47.3	66.0	69.2	65.1	89.9

## 5 Conclusion

In this paper, we propose a statutes recommendation method using classification and co-occurrence between statutes. Statutes recommendation can assist judges to decide cases, provide the public with legal advice services. We first predict CoAs, then recommend statutes related to the CoA. After obtaining recommended statutes, statutes are resorted according to co-occurrence to obtain more precise results. The experimental results show that the proposed method can effectively and accurately recommend statutes compared to several baselines. ANN are more suitable for statutes recommendation than other classifiers. Recommending based on CoA is better than direct recommendation. Using co-occurrence can further improve effectiveness.

In the future work, we will try different representations of judgment documents. Statutes are texts too and we will consider how to use the information of the statutes themselves to better recommend statutes. Besides these, we will consider how to recommend specific statutes instead of the top  $k$ .

**Acknowledgment.** This work was supported by the National Key R&D Program of China (2016YFC0800803).


## References

1. Wang, Y., Wang, S., Stash, N., Aroyo, L., Schreiber, G.: Enhancing content-based recommendation with the task model of classification. In: Cimiano, P., Pinto, H.S. (eds.) EKAW 2010. LNCS (LNAI), vol. 6317, pp. 431–440. Springer, Heidelberg (2010). [https://doi.org/10.1007/978-3-642-16438-5\\_33](https://doi.org/10.1007/978-3-642-16438-5_33)
2. Liu, L., Lecue, F., Mehandjiev, N.: Semantic content-based recommendation of software services using context. *ACM Trans. Web* **7**(3), 17:1–17:20 (2013)
3. Linden, G., Smith, B., York, J.: Amazon.com recommendations: item-to-item collaborative filtering. *IEEE Internet Comput.* **7**(1), 76–80 (2003)
4. Ghazarian, S., Nematbakhsh, M.A.: Enhancing memory-based collaborative filtering for group recommender systems. *Expert Syst. Appl.* **42**(7), 3801–3812 (2015)
5. Hofmann, T.: Latent semantic models for collaborative filtering. *ACM Trans. Inf. Syst.* **22**(1), 89–115 (2013)
6. Yu, Y., Wan, X., Zhou, X.: User embedding for scholarly microblog recommendation. In: 54th Annual Meeting of the Association for Computational Linguistics, pp. 449–453. ACL, Berlin, Germany (2016)

7. Wang, Z., Zhang, Y.: Opinion recommendation using neural memory mode. In: 2017 Conference on Empirical Methods in Natural Language Processing, pp. 1627–1638. ACL, Copenhagen, Denmark (2017)
8. Liu, Y.H., Chen, Y.L., Ho, W.L.: Predicting associated statutes for legal problems. *Inf. Process. Manage.* **51**(1), 194–211 (2015)
9. Suykens, J.A.K., Vandewalle, J.: *Least Squares Support Vector Machine Classifiers*. Kluwer Academic Publishers, Boston (1999)



# Robust and Real-Time Face Swapping Based on Face Segmentation and CANDIDE-3

Haosen Wang<sup>(✉)</sup> , Dongliang Xie , and Lu Wei 

Institute of Network Technology,  
Beijing University of Posts and Telecommunications, Beijing, China  
920383459@qq.com, xiedl@bupt.edu.cn, weilu544@gmail.com

**Abstract.** Despite some successes have been made in face swapping research, face swapping is still not robust and real-time enough. In this paper, a robust and real-time method for face swapping based on face segmentation and CANDIDE-3 is proposed. We implement our method through four steps: face detection, face alignment, modelling and swapping. We test our method on three publicly available datasets and some videos, and results show that our method can effectively improve the robust and real-time performance of face swapping.

**Keywords:** Face swapping · Face segmentation · CANDIDE-3 model

## 1 Introduction

Face swapping means transferring a face from a source image onto a face in a target image or a video, attempting to generate a realistic and visually appealing result. Some successes have been made but some issues are still challenging due to complex facial geometry as well as our perceptual sensitivity. Face swapping requires robust and effective methods for face detection, face segmentation, pose and expression estimation and so on.

We design and implement an effective face-swapping pipeline. We use a face in a source image as  $F_S$  to replace a face in a target image as  $F_T$ . We apply face alignment in both  $F_S$  and  $F_T$ , and use the facial landmarks of  $F_S$  to generate a facial model. In the meanwhile, poses and expression parameters will be estimated. For swapping, we use the poses and expression parameters to adjust the model, then apply colour transfer and blending algorithms to adjust colour and texture of the replaced area.

The technological contributions can be summarized as follows:

- The first near-real-time pipeline for face swapping based on lightweight face segmentation and 3D model, which can be run over 7 fps on low-level computers that like an Intel Core i5 4590 computer without GPU;
- A visually appealing performance based on our robust pipeline for face swapping.

The remainder of this paper is organized as follows. In Sect. 2, we briefly review the related work. In Sect. 3, we give a detailed description of our method. Experimental results are presented in Sect. 4. Finally, our conclusion is summarised in Sect. 5.

## 2 Related Work

### 2.1 Face Swapping

Face swapping has been considered in a variety of scenarios, including animation, expression transfer and privacy protection. Blanz et al. [1] fitted a morphable model to faces and rendered the source face with the parameters of the target. Bitouk et al. [2] described an automatic face swapping using a large database of faces. Such source images are those which share similar pose and expression with the targets. Blanz et al. [1] and Bitouk et al. [2] are only for images, and the methods are not robust enough that they have some limitations such as small-angle requirement. Though our method can be applied in similar way, our method is also can be used in videos and our experiments focus on harder situations that the swapped images are randomly selected without the limitation of angles. Yang et al. [3] used optical flow to replace face expressions. They have a visually appealing performance, but the method is a bit complex. Compared with the above methods, we pay more attention on the lightweight design of face swapping that can be real-time on videos.

### 2.2 Face Modelling

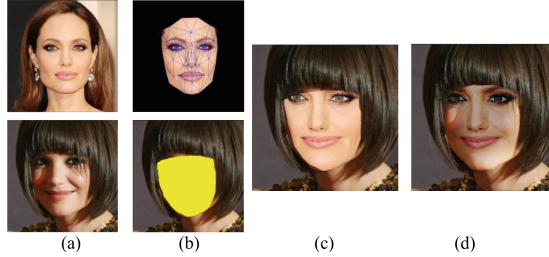
Most previous methods estimate the structure of faces based on models. Blanz et al. [1] captured model by laser scans and multi-view stereo approaches, which are quite expensive and complex. In order to map textures across different facial shapes and poses, other methods modelling from an image or multi-images estimate facial models. Bitouk et al. and Lin et al. [2, 4] used 3D Morphable Face Models (3DMM) to fit image texture. Nirkin et al. [5] used CNN to model the 3DMM model. Others [6, 7] estimated 2D active appearance models. However, the above methods that use complex 2D or 3D models are a bit time-consuming, which can not be real-time.

### 2.3 Face Segmentation

To only swap faces without their surrounding occlusions, we require segmenting faces in pixel level. Liu et al. [8] segmented facial region but not the entire face. Qian et al. [9] used face colour detection to segment the face area, but the method was not robust enough. Recently [5] used fully convolutional network (FCN) for segmentation, which led to a robust performance. Thus, we adopt FCN to improve face swapping.

### 3 System Design

Figure 1 summarizes our pipeline for face swapping. Our method first use DLIB [10] to detect faces in both sides to get the areas of faces. And then we localize facial landmarks inspired from [11] to fit the model of  $F_S$  and estimate the parameters of poses and expression. In the meanwhile, we segment the face from backgrounds and occlusions using our FCN. Finally, the  $F_S$  is efficiently swapped onto  $F_T$  and blended into the final result.



**Fig. 1.** Overview of our method. (a) Source and target image. (b) Landmarks are used to establish the CANDIDE-3 model. Face segmentation is shown on the bottom. (c) Before blending. (d) Final result that has the same shadow with the target.

#### 3.1 Face Alignment and Modelling

**Pose Estimation.** Given a set of 2D face landmarks,  $\mathbf{p} = \{p_i\} \in \mathbb{R}^2$ , and the points of anthropometric [12],  $\mathbf{q} = \{q_i\} \in \mathbb{R}^3$ , we could get the face pose by solving the Perspective-n-Point (PNP) question. Thus, the projection of face and image plane  $p_i \leftrightarrow q_i$  could be  $\mathbf{p} = \mathbf{s}\mathbf{A}\mathbf{q}$  and  $\mathbf{A}$  is a matrix of camera parameters which can be approximately measured by the image size. After pose estimation, we get the rotation matrix  $\mathbf{R}$ , which contains the rotation vector and translation vector. Rotation parameters are the pitch, yaw and roll angles of the face, which can be used to model the CANDIDE-3 model [13].

**Face Modelling.** We use CANDIDE-3 model [13] that is lightweight enough to represent faces and expressions. CANDIDE-3 model is a parameterised face mask and its low number of polygons (113 vertices and 168 surfaces) allows fast reconstruction. In this area, there may be not other methods using CANDIDE-3 model. Specifically speaking, a face model  $\mathbf{C} \in \mathbb{R}^3$  is modelled by combining the following independent generative models:

$$\mathbf{C} = \bar{\mathbf{C}} + \mathbf{V}_S\boldsymbol{\alpha} + \mathbf{V}_E\boldsymbol{\beta} \quad (1)$$

Here, vector  $\bar{\mathbf{C}}$  is the standard face model. Matrices  $\mathbf{V}_S$  (shape) and  $\mathbf{V}_E$  (expression) are principle components obtained from standard model.  $\boldsymbol{\alpha}$  and  $\boldsymbol{\beta}$  are

parameter vectors estimated by images through pose parameters. To estimate per-image facial shapes, we iterate several times to minimize the equation:

$$L = f(\mathbf{C}) - \mathbf{p} \quad (2)$$

$f(\mathbf{C})$  is a function projecting 3D matrix to 2D matrix.

### 3.2 Face Segmentation

Our method use the standard FCN to segment faces. We show segmentation can be improved after FCN is trained on plenty and varied data.

**FCN Architecture.** Long et al. [14] trained a pixels-to-pixels model on semantic segmentation that exceeds the state-of-the-art. We choose the model for our face segmentation, because we think the tasks between semantic segmentation and our face segmentation are similar, which should be a efficient and proper method.

**Training Data.** Since the FCN in [14] is not designed for face segmentation, we produce large quantities of labelled face images in open datasets. Similar to [5], we use the labelled 507 faces in COFW [15] as the training dataset. Not only that, we additionally add the faces (300 indoors and 300 outdoors) in difficult conditions in 300 W [16] and more than 13,000 faces in LFW [17] for more accurate results.

### 3.3 Face Swapping and Blending

**Face Swapping.** We use the pose and expression parameters of  $F_T$  to map the model of  $F_S$  onto the  $F_T$  area. Then we transfer the colour of  $F_S$  and render the model onto  $F_T$ .

**Colour Transfer.** Reinhard et al. [18] designed a colour transfer algorithm by representing colour in  $l\alpha\beta$  colour space. We implement the colour transfer through equation:

$$I = \frac{\sigma_T}{\sigma_S}(F_S - \text{mean}(F_S)) + \text{mean}(F_T) \quad (3)$$

Here,  $I$  is the result image.  $\sigma$  represents the standard deviation of three channels in  $(l, \alpha, \beta)$  colour space, and  $\text{mean}()$  is the mean of colour channels.

**Blending.** With Poisson fusion algorithm, we could transfer the source-texture trend through matching the gradient of target face. Minimizing the gradient diff of  $F_S$  and  $F_T$  can lead to the final result.

## 4 Experiments

We perform experiments to test our method, both qualitatively and quantitatively. Our face swapping is implemented using Caffe [19] for segmentation, DLIB [10] for face detection and facial landmarks detection and OpenCV [20] for all other image processing. Runtime is measured on an i5 4590 computer with 8GB RAM and an E5-2600 server with 16 GB RAM and a Titan X Pascal. For images, We first test our face segmentation method on COFW [15] and 300W [16] datasets, and compare with [5, 8, 21, 22]. Then we test our face-swapping method on COFW [15], 300W [16], LFW [17], and compare with [4]. For videos, almost no other methods are introduced to solve it, so we can't compare with others methods. Similar with our images evaluation, we adopt a subjective method.

### 4.1 Face Segmentation Results

Figure 2 is some results of face segmentation. We follow the evaluation procedure described by [22] and compare with other methods like [8, 22]. We use the standard intersection over union (IOU), overall percent of correctly labeled pixels(global) and the average face pixel recall(ave(face)) metrics. Table 1 shows results along with runtimes. The experiment shows that our method evaluations in IOU and global are highest and ave(face) is almost nearing the state of the art. It is noteworthy that the result of [5] is run using a higher-performance equipment. Thus, our runtime is a bit slower than the state of the art.



Fig. 2. The results of face segmentation in 300 W (1,2) and COFW (3,4).

Table 1. COFW segmentation results.

Method	Mean IOU	Global	Ave(face)	FPS
Struct.Forest [21]	-	83.9	88.6	-
SAPM [22]	83.5	88.6	87.1	-
Liu et al. [8]	72.9	79.8	89.9	0.29
Nirkin et al. [5]	83.7	88.8	<b>94.1</b>	<b>48.6</b>
Ours-PC	79.3	84.2	89.8	15.43
Ours-Server	<b>84.1</b>	<b>88.9</b>	93.2	37.3

### 4.2 Face-Swapping Results

For images, we test our method on three datasets, and also compare with [4,5] using some examples like Fig. 3. For videos, we test our face swapping using images of famous people and videos of ourselves.

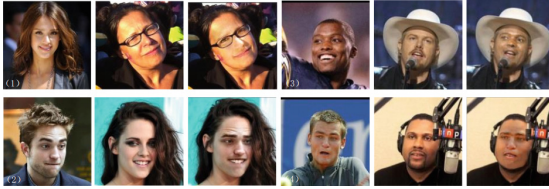


Fig. 3. The acceptable results of face swapping in 300 W (1,2) and LFW (3,4).

**Acceptable Images Examples.** We test more than 14,000 images, and swap faces in random. Then we assume the successfully detected and aligned images in swapped images are successfully swapped. The successful rate equation is described below:

$$r = \frac{w_1}{w_2} \tag{4}$$

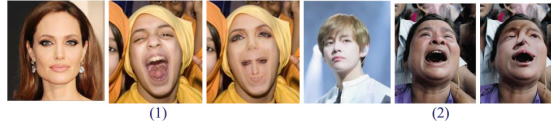
Here,  $w_1$  is the successfully swapped images (we assume), and  $w_2$  represents the whole number. As a result, our successful rate  $r$  is 87.9%. But in the successfully swapped images that we assume they are all visually appealing, some images like Fig. 4 are not acceptable that they are not visually appealing. So we manually count the acceptable images that like Fig. 3, and our acceptable rate  $a$  is 92.3% based on our successful rate  $r$ , which proves the robustness and accuracy of our method. We don't have ground-truth for the swapped face. The more possible and better way to evaluate is still subjective. Similar with [4], we randomly choose thirty acceptable images like Fig. 3, and twenty testers are asked to give scores with 1–5. 5 points indicate the wonderful performance, and 1 point indicates the poor performance. The average scores are shown in Table 2. Despite the highest score and lowest score, the average score is 4.167 points, which shows the visually appealing performance of our method.

Table 2. The average scores of twenty testers for thirty face-swapping images results.

Tester	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Score	3.9	4.2	4.3	4.1	4.7	4.6	4.1	3.6	4.2	3.9	4.2	4.3	4.4	4.1	4.4	4.1	4.0	3.8	3.9	4.5

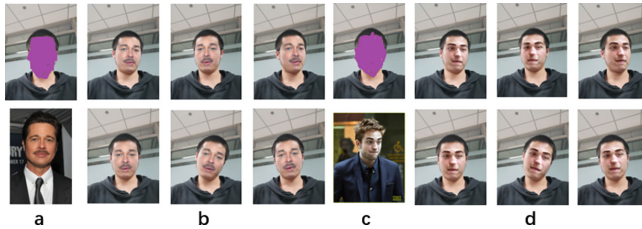
**Failed Images Examples.** We do have some failed examples as shown in Fig. 4 since wrongly detected faces will affect our results. Also, some unsuitable angles and low texture will led to unnatural results.





**Fig. 4.** Face-swapping failures. (1) Failure in detection. (2) Failure in modelling.

**Videos Results.** Some screenshots of video results are shown in Fig. 5. We randomly choose five videos like Fig. 5, and aforementioned testers are asked to give scores with 1–5, which are shown in Table 3. Despite the highest score and lowest score, the average score is 4.533 points. The scores of videos results are much higher than images scores, because the definition of our videos is higher than images from datasets. We also calculate the speed of our overall pipeline for videos, and the average speed is about 7.33 fps, which is almost real-time that people can be acceptable in the face-swapping application.



**Fig. 5.** Some screenshots of the face-swapping results. Label a and c include images of ourselves in videos (Faces masked for privacy concerns) and famous persons. b and d are swapped results.

**Table 3.** The average scores of twenty testers for five face-swapping videos results.

Tester	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Score	4.7	4.4	4.5	4.6	4.7	4.6	4.8	4.2	4.4	4.6	4.5	4.7	4.6	4.5	4.4	4.6	4.1	4.7	4.4	4.5

## 5 Conclusion

We describe a visually appealing and real-time method for face swapping which is accurate and robust enough to allow large-scale tests. To improve our method, our future work will consider end-to-end methods using deep learning.

## References

1. Blanz, V., Scherbaum, K., Vetter, T., Seidel, H.P.: Exchanging faces in images. In: Computer Graphics Forum, pp. 669–676 (2004)
2. Bitouk, D., Kumar, N., Dhillon, S., Belhumeur, P., Nayar, S.K.: Face swapping: automatically replacing faces in photographs. In: ACM SIGGRAPH, p. 39 (2008)

3. Yang, F., Wang, J., Shechtman, E., Bourdev, L., Metaxas, D.: Expression flow for 3D-aware face component transfer. *Acm Trans. Graph.* **30**(4), 1–10 (2011)
4. Lin, Y., Wang, S., Lin, Q., Tang, F.: Face swapping under large pose variations: a 3D model based approach. In: 2012 IEEE International Conference on Multimedia and Expo (ICME), pp. 333–338. IEEE (2012)
5. Nirkin, Y., Masi, I., Tran, A.T., Hassner, T., Medioni, G.: On face segmentation, face swapping, and face perception. arXiv preprint [arXiv:1704.06729](https://arxiv.org/abs/1704.06729) (2017)
6. De La Hunty, M., Asthana, A., Goecke, R.: Linear facial expression transfer with active appearance models. In: 2010 20th International Conference on Pattern Recognition (ICPR), pp. 3789–3792. IEEE (2010)
7. Zhu, J., Van Gool, L., Hoi, S.C.: Unsupervised face alignment by robust nonrigid mapping. In: 2009 IEEE 12th International Conference on Computer Vision, pp. 1265–1272. IEEE (2009)
8. Liu, S., Yang, J., Huang, C., Yang, M.H.: Multi-objective convolutional learning for face labeling. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3451–3459 (2015)
9. Qian, K., Wang, B., Chen, H.: Automatic flexible face replacement with no auxiliary data. *Comput. Graph.* **45**, 64–74 (2014)
10. King, D.E.: Dlib-ml: a machine learning toolkit. *J. Mach. Learn. Res.* **10**, 1755–1758 (2009)
11. Kazemi, V., Sullivan, J.: One millisecond face alignment with an ensemble of regression trees. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1867–1874 (2014)
12. Nguyen, A., Simard-Meilleur, A., Berthiaume, C., Godbout, R., Mottron, L.: Head circumference in canadian male adults: development of a normalized chart. *Int. J. Morphol.* **30**(4), 1474–1480 (2012)
13. Ahlberg, J.: CANDIDE-3 - an updated parameterised face. *Rinsho Byori Jpn. J. Clin. Pathol.* **48**(3), 385–388 (2001)
14. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Computer Vision and Pattern Recognition, pp. 3431–3440 (2015)
15. Burgos-Artizzu, X.P., Perona, P., Dollar, P.: Robust face landmark estimation under occlusion. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1513–1520 (2013)
16. Sagonas, C., Tzimiropoulos, G., Zafeiriou, S., Pantic, M.: 300 faces in-the-wild challenge: the first facial landmark localization challenge. In: IEEE International Conference on Computer Vision Workshops, pp. 397–403 (2014)
17. Learned-Miller, E., Huang, G.B., Roychowdhury, A., Li, H., Hua, G.: Labeled Faces in the Wild: A Survey. Springer International Publishing, Heidelberg (2016)
18. Reinhard, E., Adhikmin, M., Gooch, B., Shirley, P.: Color transfer between images. *IEEE Comput. Graph. Appl.* **21**(5), 34–41 (2001)
19. Jia, Y., et al.: Caffe: convolutional architecture for fast feature embedding. In: Proceedings of the 22nd ACM International Conference on Multimedia, pp. 675–678. ACM (2014)
20. Bradski, G.: The opencv library. *Dr. Dobbs J. Softw. Tools Prof. Program.* **25**(11), 120–123 (2000)
21. Jia, X., Yang, H., Chan, K.P., Patras, I.: Structured semi-supervised forest for facial landmarks localization with face mask reasoning. In: BMVC (2014)
22. Ghiasi, G., Fowlkes, C.: Using segmentation to predict the absence of occluded parts. In: British Machine Vision Conference, pp. 22.1–22.12 (2015)



# Determining the Applicability of Advice for Efficient Multi-Agent Reinforcement Learning

Yuchen Wang<sup>(✉)</sup>, Fenghui Ren, and Minjie Zhang

School of Computing and Information Technology, University of Wollongong,  
Wollongong, NSW 2522, Australia  
yw808@uowmail.edu.au, {fren,minjie}@uow.edu.au

**Abstract.** Action advice is an important mechanism to improve the learning speed of multiple agents. To do so, an advisor agent suggests actions to an advisee agent. In the current advising approaches, the advisor's advice is always applicable based on the assumption that the advisor and advisee have the same objective, and the environment is stable. However, in many real-world applications, the advisor and advisee may have different objectives, and the environment may be dynamic. This would make the advisor's advice not always applicable. In this paper, we propose an approach where the advisor and advisee jointly determine the applicability of advice by considering the different objectives and dynamic changes in the environment. The proposed approach is evaluated in various robot navigation domains. The evaluation results show that the proposed approach can determine the applicability of advice. The multi-agent learning speed can also be improved benefiting from determined applicable advice.

**Keywords:** Action advice · Multi-agent reinforcement learning  
Different objectives · Dynamic environment

## 1 Introduction

Multi-agent reinforcement learning (MARL) is an important technique for agents to learn to achieve the agents' goals in a multi-agent environment [2]. An environment can be represented by a set of states. A typical goal of an agent is to reach a specific objective state. MARL methods are known to take a long time to reach convergence [6]. An important mechanism to speed up learning is action advice [7] where an experienced agent (advisor) gives advised actions (advice) to a less experienced agent (advisee). Applicable advice can help the advisee to reach its objective from the state that the advisee is in.

This paper focuses on the problem of how to determine the *applicability* of the advisor's advice. The advice for a state is applicable when the advice can speed up the agents' learning. In current literature, an advisor can give applicable

advice for all states. However, two factors would make the advice not applicable to some particular states. The first factor is the difference of objectives. When the advisor and advisee have different objectives, the advice may not be applicable because the advice is not intended for reaching the advisee’s objective. The second factor is the dynamic changes in the environment. When the current environment is different from the environment where the advisor learns its experience, the advice may not be applicable to this new environment. These factors exist in real-world applications. For example, suppose a robot  $a$  is learning to navigate to its objective in an environment. After  $a$  has learned a good policy for navigating to its objective (thus  $a$  acts as an advisor), another robot  $b$  with no learning experience joins the environment (thus  $b$  acts as an advisee).  $b$  has a different objective from  $a$ . In this example, the environment now has changed from a single-agent environment to a multi-agent environment. In this situation, the advice for some particular states may not be applicable to the advisee due to the difference of objectives and the change of the environment. Hence, how can the advisor advise the advisee with different objectives in a dynamically changing environment is a significant research challenge of MARL.

Many advising approaches have been proposed [1, 4, 5, 7, 10, 12]. For example, Maclin and Shavlik [4] designed an advising approach which allowed an external advisor to give advice to learning advisees. Torrey *et al.* [7] proposed a teacher-student approach which limited the number of times the teacher could provide advice. However, these approaches require that the advisor and advisee have the same objective. Also, the environment is assumed to be stable. Therefore, the existing advising approaches cannot handle the advice applicability problem we consider.

Against this background, in this paper, we propose an approach named JDAA (stands for Joint Determination of Applicability of Advice). In this approach, the advisor and advisee jointly determine the applicability of the advisor’s advice. We introduce the concept of the applicability of advice by considering two factors, i.e., the difference of objectives and the dynamic changes in the environment. When the advisee asks the advisor for advice in a state  $s$ , the advisor determines whether its advice is *helpful* to reach the advisee’s objective from  $s$  by considering the difference of objectives. The advisee determines whether the advice for  $s$  is *acceptable* by considering the dynamic changes in the environment. The advice is determined to be applicable if the advisor and advisee determine that the advice is helpful and acceptable respectively. For testing purposes, the proposed approach is employed to the robot navigation domains. Experimental results show that using the proposed approach, the advisor and advisee can jointly determine the applicability of advice. Benefiting from the determined applicable advice, the multi-agent learning speed can be improved.

The remainder of this paper is organised as follows. Section 2 introduces how the advisor and advisee jointly determine the applicability of advice. Section 3 shows the experimental evaluations. Section 4 concludes this paper.

## 2 Joint Determination of Applicability of Advice

We hereby introduce the concept of the applicability of the advisor’s advice. The advisee asks the advisor for advice in a state. The advice for this state is said to be applicable if the advice is *helpful* from the advisor’s perspective, and *acceptable* from the advisee’s perspective. The advice is determined to be helpful if the advisor considers that the advice can help the advisee to learn. The advice is determined to be acceptable if the advisee confirms that the advice will not cause significant negative reward.

We propose an approach where the advisor and advisee jointly determine the applicability of the advisor’s advice. In this approach, the advisor determines whether the advice can help the advisee to reach its objective. The difference of objectives between the advisor and advisee is considered in the advisor’s decision. The advisee determines whether the advice is acceptable. The dynamic changes in the environment are considered in the advisee’s decision. The determination process is described as follows. At the beginning, the advisee observes its current state  $s$ . The advisor needs to determine whether the advice for  $s$  is helpful to reach the advisee’s objective by considering the difference of objectives. To do so, the advisee generates some sub-objective states  $S_{sub.obj(ee)}$ . The advisee asks the advisor for advice by sending  $s$  and  $S_{sub.obj(ee)}$  to the advisor. The advisor then generates an advised action  $\pi_{or}(s)$ , and determines whether  $\pi_{or}(s)$  is helpful for the advisee to reach one of  $S_{sub.obj(ee)}$  from  $s$ . If  $\pi_{or}(s)$  is determined to be helpful, the advisor suggests  $\pi_{or}(s)$  to the advisee. Then, the advisee determines whether  $\pi_{or}(s)$  is acceptable by considering the dynamic changes in the environment. If  $\pi_{or}(s)$  is determined to be acceptable,  $\pi_{or}(s)$  is jointly determined to be applicable. The advisee then adopts  $\pi_{or}(s)$  as the next action to execute. If  $\pi_{or}(s)$  is determined to be not applicable, the advisee uses existing action selection methods to select the next action.

The generation of the advised action  $\pi_{or}(s)$  and related concepts are introduced in Sect. 2.1. Section 2.2 introduces the definition of the sub-objective, and describes how the advisor determines whether  $\pi_{or}(s)$  is helpful for the advisee to reach one of its sub-objectives. Section 2.3 describes how the advisee determines whether  $\pi_{or}(s)$  is acceptable.

### 2.1 Generating Advised Action

Action advice [7] is used as the basic advising framework. In this framework, the advisor’s advice is represented as an action. The advisor’s learned experience is presented as a policy  $\pi_{or}$ . For a state  $s$ ,  $\pi_{or}(s)$  indicates the action to execute to reach the advisor’s objective from the state  $s$ . When the advisee asks the advisor for advice in the state  $s$ , the advised action can be generated by  $\pi_{or}(s)$ . Although  $\pi_{or}(s)$  is intended for reaching the advisor’s objective,  $\pi_{or}(s)$  could be helpful to reach the advisee’s objective if  $\pi_{or}(s)$  is determined to be applicable.

## 2.2 Determining Whether the Advised Action Is Helpful

After generating the advised action  $\pi_{or}(s)$ , the advisor needs to determine whether  $\pi_{or}(s)$  is helpful to reach one of the advisee's sub-objectives  $S_{sub\_obj(ee)}$  from the state  $s$ . This determination can be made using the equations below:

$$Helpful(s, \pi_{or}(s)) = \mathbb{1}[\exists s_{sub\_obj(ee)}^*(s, \pi_{or}(s))], \quad (1)$$

$$s_{sub\_obj(ee)}^*(s, \pi_{or}(s)) = \arg \max_{s'} \{V_{ee}(s') | s' \in S_{sub\_obj(ee)} \wedge s' \in SV(s, \pi_{or}(s))\}. \quad (2)$$

where  $V_{ee}$  is the value function of the advisee,  $SV(s, \pi_{or}(s))$  is a state set which contains the most possible states to meet when the advisor travels from the state  $s$  to the advisor's objective,  $S_{sub\_obj(ee)}$  is a state set which contains the sub-objectives of the advisee,  $s_{sub\_obj(ee)}^*$  is the advisee's most valuable sub-objective that is in both  $S_{sub\_obj(ee)}$  and  $SV(s, \pi_{or}(s))$ . If  $s_{sub\_obj(ee)}^*$  exists, the advised action  $\pi_{or}(s)$  is determined to be helpful.

## 2.3 Determining Whether Advised Action Is Acceptable

In a dynamically changing environment, the advisor and advisee cannot anticipate whether the advised action  $\pi_{or}(s)$  would cause significant negative reward (e.g., caused by a collision with another agent). Hence, after receiving  $\pi_{or}(s)$ , the advisee needs to determine whether  $\pi_{or}(s)$  is acceptable. This determination can be made using the equations below:

$$Acceptable(s, \pi(s)) = \mathbb{1}[P_{accept}(s, \pi_{or}(s)) > P_{min}], \quad (3)$$

$$P_{accept}(s, \pi_{or}(s)) = \frac{1}{1 + e^{n(s, \pi_{or}(s))}}. \quad (4)$$

where  $P_{accept}(s, \pi_{or}(s))$  indicates the possibility of accepting  $\pi_{or}(s)$  in the state  $s$ ,  $P_{min}$  is a threshold for determining whether  $\pi_{or}(s)$  is acceptable. If  $P_{accept}(s, \pi_{or}(s))$  is greater than  $P_{min}$ , the advised action  $\pi_{or}(s)$  can be accepted for the state  $s$ .  $P_{min}$  can be set to a small value (e.g., 0.01).  $P_{accept}(s, \pi_{or}(s))$  is computed by a logistic function. In this function,  $n(s, \pi_{or}(s))$  is the number of times that  $\pi_{or}(s)$  causes significant negative reward in the state  $s$ .  $n(s, \pi_{or}(s))$  will be updated during learning to calculate the value of  $P_{accept}(s, \pi_{or}(s))$ .

## 3 Experimental Evaluation

We conduct two experiments to evaluate the proposed approach. The experimental results and analysis are demonstrated. For all experiments, three kinds of advising approaches are performed: (1) No-Advice; (2) the proposed JDAA approach; and (3) the state-of-the-art Teacher-Student (TS) approach [7].

### 3.1 Experimental Setup

**Basic Settings.** The experiments are conducted in various robot navigation domains, which have been widely used to study MARL problems [3, 8, 9, 11]. In each of these domains, each robot tries to navigate to its objective state. The state of a robot is its location, which is represented in a coordinate system. The domains used in the experiments are shown in Fig. 1. The parameters used in the learning algorithm and the proposed approach are shown in Table 1.

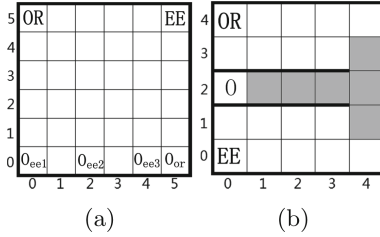


Fig. 1. Robot navigation domains. (a) SQUARE. (b) TTG.

**Two Experiments.** We conduct two experiments to show (a) the determination results of the applicability of advice; and (b) the improvement on the speed of learning benefiting from the determined applicable advice.

**Experiment 1.** The first experiment is conducted under various cases specific to the difference of objectives. Figure 1(a) shows the domain used in this experiment. The initial states of the advisor and advisee are plotted as “OR” and

Table 1. Parameters setting

Parameters	Values	Meanings
$\alpha$	0.02	The learning rate
$\gamma$	0.99	The discount factor
$\epsilon$	0.01	The exploration factor
$r$	+200	The reward of reaching objective
	-10	The reward of colliding with walls or other agents
	-1	The reward of executing an action
$N$	5000	The number of learning episodes
$\tau$	0	The threshold for generating sub-objectives
$P_{min}$	0.01	The threshold for determining
		whether the advice is acceptable

Table 2. The settings for Experiment 1 and Experiment 2

The setting for Experiment 1		The setting for Experiment 2	
Cases	The advisee's objective state	Cases	The probability of failure of the advisor's actions in shaded states
Case 1	(0, 0)	Case 1	0
Case 2	(2, 0)	Case 2	0.4
Case 3	(4, 0)	Case 3	0.8

“EE” respectively. The objective states of the advisor and advisee are plotted as “ $O_{or}$ ” and “ $O_{ee}$ ” respectively. The advisee’s objective is chosen from three states shown in Table 2 (left). Each state indicates one case of the experiment.

**Experiment 2.** The second experiment is conducted under various cases specific to the changing environment. Figure 1(b) shows the domain used in this experiment. The objectives of the advisor and advisee are the same (the state (0, 2)). In the shaded states, the advisor’s action is assigned a probability of failure which makes the advisor’s state unchanged. The probability of failure is chosen from three values shown in Table 2 (right). Also, in the shaded states, a collision happens if the agents pass simultaneously. In other states, the agents can move freely.

### 3.2 Experimental Results and Analysis for Different Objectives

Figure 2(a)–(c) show the determination results of the applicability of advice in Experiment 1. For a state  $s$ , the  $s_{sub\_obj(ee)}^*(s, \pi_{or}(s))$  (refer to Eq. (2)) is plotted on  $s$  if the advice for  $s$  is applicable. Otherwise, “N/A” (stands for Not Applicable) is plotted. We can see that the proposed approach can determine the applicability of advice for each state. The difference of objectives will influence the value of  $s_{sub\_obj(ee)}^*(s, \pi_{or}(s))$  for each state.

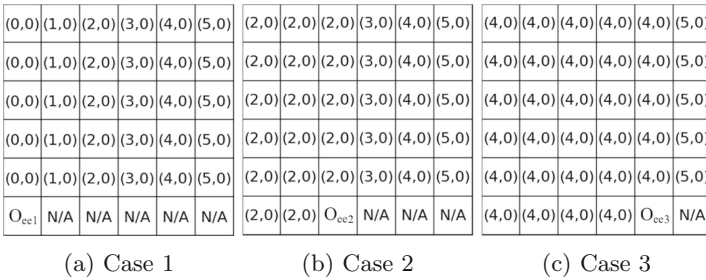


Fig. 2. The applicability of advice for each state in the cases of Experiment 1.

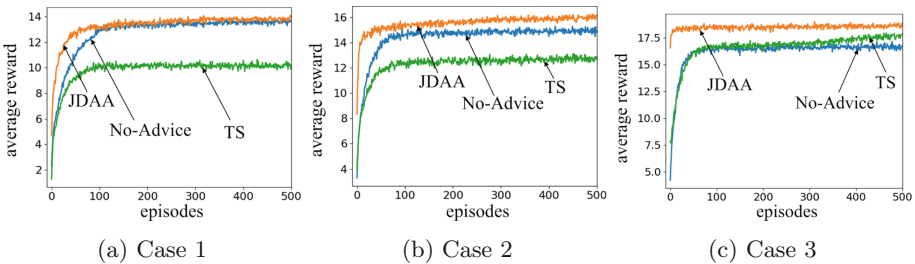


Fig. 3. Learning dynamics of three advising approaches in Experiment 1.



Figure 3(a)–(c) show the learning dynamics of the three advising approaches in Experiment 1. In Case 1, although JDAA and No-Advice finally converge to similar average reward, JDAA can increase the learning speed in the early stage of learning. In Case 2, the advisee’s objective becomes “closer” to the advisor’s objective compared with Case 1. JDAA achieves better asymptotic performance and higher total reward than No-Advice and TS. However, in Case 1 and 2, TS performs worse than JDAA and No-Advice. This is because the advice of TS is based on the advisor’s objective, which misleads the advisee. In Case 3, JDAA performs even better than No-Advice. Interestingly, TS also performs better than No-Advice. This makes sense because in this case, the advisee’s objective is “very close” to the advisor’s objective. Overall, JDAA outperforms No-Advice and TS in all cases.

### 3.3 Experimental Results and Analysis in Dynamic Environment

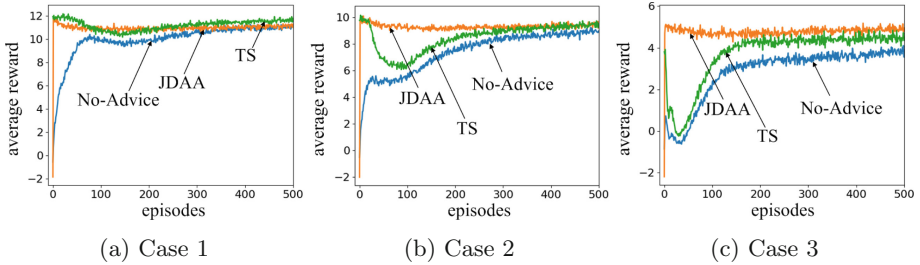
Figure 4(a)–(c) show the determination results of the applicability of advice in Experiment 2. In Case 1, the advice is not applicable to only one state near the doorway. This is because the advisor and advisee have high chance to move simultaneously to the doorway. Hence, the advised action to the advisee has high chance to collide with the advisor’s action. When the advisor’s action becomes more stochastic, the advice is not applicable for more states because the advisee is more likely to collide with the advisor in and near the shaded states.

Figure 5(a)–(c) show the learning dynamics of the three advising approaches in Experiment 2. In Case 1, TS performs slightly better than the other approaches because the advisor’s actions are not very stochastic. Both agents can keep moving towards their objectives without causing many collisions. When the advisor’s actions become more stochastic, the performance of No-Advice and TS is more negatively impacted. By contrast, the performance of JDAA is not significantly impacted because when the advisee is in states with high collision probability, the advisee does not accept the advised actions and selects “safe” actions instead. Overall, JDAA can better deal with the dynamic environment compared with No-Advice and TS.

(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)
(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)
O <sub>cc</sub>	(0,2)	(0,2)	(0,2)	(0,2)	O <sub>cc</sub>	(0,2)	(0,2)	N/A	N/A	O <sub>cc</sub>	(0,2)	N/A	N/A	N/A
(0,2)	(0,2)	(0,2)	(0,2)	N/A	(0,2)	(0,2)	(0,2)	(0,2)	N/A	(0,2)	(0,2)	(0,2)	N/A	N/A
(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	(0,2)	N/A

(a) Case 1
(b) Case 2
(c) Case 3

**Fig. 4.** The applicability of advice for each state in the cases of Experiment 2.



**Fig. 5.** Learning dynamics of three advising approaches in Environment 2.

## 4 Conclusion

In this paper, we proposed an approach where the advisor and advisee could jointly determine the applicability of the advisor’s advice. The difference of objectives and the dynamic changes in the environment had been considered as factors that might influence the advice’s applicability. Compared with the state-of-the-art work, the proposed approach could determine the applicability of advice for each state in the environment. The multi-agent learning speed could be improved benefiting from the determined applicable advice. In future, we plan to theoretically analyse how each sub-objective contributes to reaching the agent’s objective, which would bring insights into the proposed approach.

## References

1. Amir, O., Kamar, E., Kolobov, A., Grosz, B.J.: Interactive teaching strategies for agent training. In: Proceedings of the 25th International Joint Conferences on Artificial Intelligence, pp. 804–811 (2016)
2. Busoniu, L., Babuska, R., De Schutter, B.: A comprehensive survey of multiagent reinforcement learning. *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.* **38**(2), 156–172 (2008)
3. De Hauwere, Y.M., Vrancx, P., Nowé, A.: Learning what to observe in multi-agent systems. In: Proceedings of the 20th Belgian-Netherlands Conference on Artificial Intelligence, pp. 83–90. Citeseer (2009)
4. Maclin, R., Shavlik, J.W.: Creating advice-taking reinforcement learners. *Mach. Learn.* **22**(1–3), 251–281 (1996)
5. Taylor, M.E., Carboni, N., Fachantidis, A., Vlahavas, I., Torrey, L.: Reinforcement learning agents providing advice in complex video games. *Connect. Sci.* **26**(1), 45–63 (2014)
6. Taylor, M.E., Stone, P.: Transfer learning for reinforcement learning domains: a survey. *J. Mach. Learn. Res.* **10**, 1633–1685 (2009)
7. Torrey, L., Taylor, M.: Teaching on a budget: agents advising agents in reinforcement learning. In: Proceedings of the 12th International Conference on Autonomous Agents and Multiagent systems, pp. 1053–1060 (2013)
8. Yu, C., Zhang, M., Ren, F.: Coordinated learning by exploiting sparse interaction in multiagent systems. *Concurrency Comput. Pract. Experience* **26**(1), 51–70 (2014)

9. Yu, C., Zhang, M., Ren, F., Tan, G.: Multiagent learning of coordination in loosely coupled multiagent systems. *IEEE Trans. Cybern.* **45**(12), 2853–2867 (2015)
10. Zhan, Y., Ammar, H.B., Taylor, M.E.: Theoretically-grounded policy advice from multiple teachers in reinforcement learning settings with applications to negative transfer. In: *Proceedings of the 25th International Joint Conference on Artificial Intelligence*, pp. 2315–2321 (2016)
11. Zhou, L., Yang, P., Chen, C., Gao, Y.: Multiagent reinforcement learning with sparse interactions by negotiation and knowledge transfer. *IEEE Trans. Cybern.* **47**(5), 1238–1250 (2017)
12. Zimmer, M., Viappiani, P., Weng, P.: Teacher-student framework: a reinforcement learning approach. In: *AAMAS Workshop Autonomous Robots and Multirobot Systems* (2014)



# Multi-object Detection Based on Deep Learning in Real Classrooms

Benchi Shao, Fei Jiang, and Ruimin Shen<sup>(✉)</sup>

Department of Computer Science and Engineering, Shanghai Jiao Tong University,  
800 Dongchuan Rd, Shanghai, China  
{shaobenchi, jiangf, rshen}@sjtu.edu.cn

**Abstract.** Information-based classrooms with cameras provide numerous videos for evaluating teaching qualities. In this paper, we focus on automatically detecting the learning-and-teaching related behaviors for further teaching analysis, including standing-up and hand-raising of students, and movements of teachers. First, due to the continuity of behaviors, we convert it into a frame-based object detection problem. Compared with the publicly-available datasets of object detection, there are several challenges in our real classrooms, such as low resolutions, various gestures, complex backgrounds, and occlusions. Second, to solve these challenges, we propose an improved R-FCN architecture, which incorporates Feature Pyramid Networks into Region Proposal Networks (RPNs) and introduces a position-sensitive RoIAlign layer. The multi-level features and RPNs provide more contexture information for small object detections, and the position-sensitive RoIAlign layer reduces the misalignment in extracting features for region proposals. Lastly, the efficiency of the proposed algorithm is demonstrated on our collected data from real classrooms, which contains 30k frames with 100k labeled objects.

**Keywords:** Multi-object detection · Deep learning · Classroom

## 1 Introduction

Along with the advancement of information-based classrooms in primary and middle schools, numerous videos that record the behaviors of students and teachers are available, which can be used for analyzing the teaching qualities in the traditional education. However, most existing teaching quality evaluations are based on labors, which are time consuming and expensive. In this paper, we focus on automatically detecting the learning-and-teaching related behaviors for further teaching analysis. The learning-and-teaching behaviors include stand-up and hand-raising of students, and the movements of teachers, which can be used to evaluate the activeness of teaching. Due to the continuity of behaviors, the related behaviors detection is converted into a frame-based multi-object detection, as shown in Fig. 1. Compared with the publicly-available datasets of object

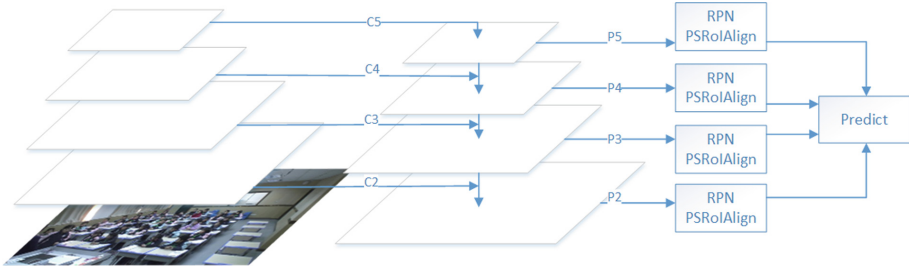


**Fig. 1.** The learning-and-teaching related behaviors detection is converted into a frame-based standing (blue box), hand-raising (orange boxes) and teacher (green box) multi-object detection. (Color figure online)

detection, the object detection in real classrooms are quite challenging due to low resolutions, various gestures, complex backgrounds, and occlusions.

Nowadays, one popular strategy in object detection algorithms is based on region proposals. Region-based systems consist of three main steps: (1) generate a certain number of region proposals, (2) extract features for each region proposal, (3) use a classifier to estimate the category for each region proposal. Several region proposal based algorithms have received the state-of-the-art results on publicly-available datasets, such as R-CNN [5], Fast R-CNN [4], Faster R-CNN [14] and R-FCN [2]. R-CNN is to attend to a number of region proposals and evaluates convolutional networks on each candidate region. Fast R-CNN [4] accelerates R-CNN by sharing convolutional layers among region proposals. Moreover, Faster R-CNN [14] introduces Region Proposal Network (RPN) which computes region proposals using full convolutional network and reduces the cost of generating proposals. R-FCN removes the costly RoI-wise subnetwork and shares all learnable layers, achieving competitive results and reducing test-time compared to Faster R-CNN. Therefore, we focus on the R-FCN architecture and propose an improved R-FCN algorithm for our real application.

Due to the complexity of real applications, in this paper, we propose an improved R-FCN algorithm for the multi-object detection in real classrooms. As shown in Fig. 1, the scale of objects in different positions varies because of their different distances to the camera. First, to address the multi-scale problem in real classrooms, we utilize FPN [11] to obtain multi-level feature maps with different scales. Second, we introduce a position-sensitive RoIAlign layer which incorporates alignment to a position-sensitive RoI pooling layer. Instead of using rounding operation in calculating the boundaries of RoIs and divided bins, we keep the floating-number RoIs and use bilinear interpolation [8] to extract more accurate features for RoIs. The experiments conducted on our dataset demonstrate that our method improves the performance of multi-object detection in classroom (Fig. 2).



**Fig. 2.** Overall architecture of our method. Feature pyramid is built across  $C_2$  to  $C_5$  and RPNs is attached to the merged feature maps P2-P5.

## 2 Related Work

We briefly introduce several region proposal methods based on low-level features and Convolutional Neural Network (CNN) [9, 10]. Then, RoI pooling and RoIAlign layers are introduced for their differences in extracting features for region proposals.

### 2.1 Region Proposal Algorithms

Region proposal methods can be mainly classified into two categories. The first category is based on low-level features such as superpixel [13, 17] and edge [18]. Beside above-mentioned methods based on traditional features, the other category is using CNN to generate region proposals. DeepMultiBox [3] generates multiple candidate regions by a single CNN in a class-agnostic manner. These class-agnostic regions can be used as proposals for detectors like Fast R-CNN. Region Proposal Network (RPN) [14] adopts a fully convolutional network (FCN) [12] to generate region proposals and corresponding objective scores. The computation of convolutional layers is shared between RPN and Fast R-CNN detection backbone.

### 2.2 RoI Pooling and RoIAlign Layers

RoIAlign [6] removes the approximation operation used by RoI pooling layer in the implementation of mapping a region to feature maps. A region is defined by  $(x_1, y_1, x_2, y_2)$  that specifies its top-left corner  $(x_1, y_1)$  and down-right  $(x_2, y_2)$  in the image domain. RoI Pooling uses a rounding operation to get the correspond RoI  $([x_1/16], [y_1/16], [x_2/16], [y_2/16])$  where a feature map stride is 16 and  $[\cdot]$  is round operation. The RoI is divided into bins and the boundary of bin use the similar strategy to obtain integral values. However, RoIAlign keeps the float-number as  $x/16$  instead of  $[x/16]$  and uses bilinear interpolation to compute the value of float value points within a RoI bin, addressing the misalignment between RoI and the features extracted from feature map.

### 3 Our Method

We introduce the architecture of the proposed algorithm, and the implementation details of combining FPN with RPNs and position-sensitive RoIAlign layer. The backbone of our method is based on ResNet-101 [7] and the output of conv2, conv3, conv4, and conv5 denoted as  $\{C_2, C_3, C_4, C_5\}$ .

#### 3.1 Feature Pyramid Network with RPNs

FPN takes top-down path and lateral connection as the building block of feature pyramid. Top-down pathway upsamples higher level feature map  $C_{i+1}$  ( $2 \leq i \leq 4$ ) by a factor of 2 and nearest neighbor up-sampling is used for simplify. The lateral connection merges  $C_i$  (a  $1 \times 1$  convolutional layer is applied to reduce channel dimensions) and with the upsampled  $C_{i+1}$  by element-wise sum. This process iterates from the coarsest  $C_5$  feature map to the finest  $C_2$  level and the corresponding final feature map is  $\{P_2, P_3, P_4, P_5\}$ . For  $C_5$ , it does not have a higher layer of feature map and just undergoes a  $1 \times 1$  convolutional layer to reduce the numbers of channels to 256 (all levels of feature map in FPN have 256 channels). A  $3 \times 3$  convolution layer is appended to each merged result to obtain the final feature maps.

Our method uses each FPN level of feature map to generate region proposals and extract features for RoIs simultaneously. The RoIs generated by different RPNs are collected across all levels and those with top scoring are retained. The chosen RoIs will be distributed to each FPN level for further feature extraction.

#### 3.2 Position-Sensitive Feature Map and Position-Sensitive RoIAlign

Position-sensitive feature map is constructed to produce  $k^2 \times (C + 1)$  feature map ( $k^2$  feature maps for  $(C+1)$  classes). Given a region  $(x_1, y_1, x_2, y_2)$  defined in the image domain, the mapping RoI in feature map with stride 16 can be represented by  $(x_1, y_1, x_2, y_2)/16$ . Then each RoI is divided into  $k^2$  bins, and the  $(i, j)$ -th bin spans:

$$(x_1 + i \times \frac{x_1 - x_2}{k})/16 \leq x \leq (x_1 + (i + 1) \times \frac{x_1 - x_2}{k})/16 \quad (1)$$

$$(y_1 + i \times \frac{y_1 - y_2}{k})/16 \leq y \leq (y_1 + (i + 1) \times \frac{y_1 - y_2}{k})/16 \quad (2)$$

Position-sensitive RoIAlign layer perform average pooling operation on four regular locations within each RoI bin. We also experiment to sample adaptive number of grid points (computed by  $\lceil (x_2 - x_1)/k^2 \rceil \times \lceil (y_2 - y_1)/k^2 \rceil$ ,  $\lceil \cdot \rceil$  is ceiling), which we found to give almost same returns. The value of the sampled points is calculated by bilinear interpolation [8]. The  $k^2(C + 1)$ -dimensional pooling output then perform average pooling on each  $k^2$  position-sensitive scores, obtaining a  $(C + 1)$  length vector.

Following R-FCN, aside above  $k^2(C + 1)$ -channels Position-sensitive feature map for classification, we also construct an  $4k^2(C + 1)$ -channels convolutional layers for class-specific bounding box regression.

### 3.3 Training

For training RPNs, we follow the setting in Faster R-CNN that an anchor is labeled positive when having an intersection-over-union (IoU) with ground-truth bounding box more than 0.7, negative for IoU lower than 0.3. The loss function of reach anchor from RPN is defined as:

$$L(c, b) = L_{cls}(c, c^*) + \lambda c^* L_{loc}(b, b^*) \quad (3)$$

Here,  $c^*$  is the ground-truth label (0 means negative, 1 means positive) and  $c$  denote the possibility of containing an object.  $L_{cls}$  use binary cross-entropy to calculate the classification loss. We use the bounding box regression loss defined in [4].

We use approximate joint training [14] with momentum of 0.9 and a weight decay of 0.0001. Images are resized such that the longer side of an image is 1024 pixels. We use the ResNet-101 pre-trained on ImageNet [16]. Our model is trained on 2 GPUs (one image per GPU) for 45k iterations and an image has 1024 RoIs, with positive ratio of 25%. We use the Caffe2 deep learning framework on ubuntu 16. The base learning rate is 0.001 and decreases by 10 for 30k iteration and 40k iteration. We flip the training data to increase the number of images.

**Table 1.** The numbers of instances used in training and test

	Standing	Hand-raising	Teacher	Total
Train	10013	34365	7288	51666
Test	10058	34369	7342	51769

## 4 Experiments

Several experiments are conducted to evaluate the performances of the proposed method in our real scenario. First, a frame-based real dataset is collected, which is used for training and test the state-of-the-art detection algorithms. Then, the detection results based on average precision (AP) show the advantages of the proposed algorithm.

### 4.1 Our Dataset

Our image data comes from the cameras in front of the classrooms of primary and middle schools. As shown In Table 1, three categories of objects are annotated in our dataset with 34,710 images, including 20,071 standing instances 14,630 teacher instances and 68,734 hand-raising instances. The whole dataset is split into 50% for training and 50% for test. Our dataset has several features: fixed viewpoint, various illuminations and varied poses.



## 4.2 Experimental Details

**Faster R-CNN.** As discussed in [15], to utilize ResNet-101 in Faster R-CNN, the RPN and RoI Pooling layer share the feature of  $C_4$  and  $C_5$  become unshared, RoI-dependent feature.

**Stride and À trous.** R-FCN use the à trous trick to increase the resolution of the  $C_5$ , which make  $C_5$  has the same stride with  $C_4$ . In our experiments, we found that our method with the à trous trick achieve a relatively little improvement compared with R-FCN. The stride of  $C_5$  does not affect the performance of our method that much because the multi-level feature maps can compensate for the high stride of  $C_5$ . However, for fair comparison, we use the à trous trick in ablation experiments, denoted by ResNet-101- $C_5$  (à trous, stride = 16).

**Test.** During test phase, each image is resized to  $1024 \times 576$ . The RPNs generate RoIs and position-sensitive RoIAlign layers extract features for classification and bounding box regression. The NMS (non-maximum suppression) with a threshold of 0.5 is applied to the detection results. Meanwhile, Soft-NMS [1] with a  $\sigma$  value of 0.5 is evaluated on our dataset and achieve better performance than NMS, shown in Table 2.

**Table 2.** Comparisons between our method and Faster R-CNN (ResNet-101- $C_4$ ) and R-FCN (ResNet-101- $C_5$ , à trous, stride = 16) on our dataset. The COCO-style AP is evaluated @IoU  $\in [0.5, 0.95]$ . AP@75 is evaluated @IoU = 0.75. Our method outperforms the baseline Faster R-CNN and R-FCN. In ablations experiments, the model of position-sensitive RoIAlign is based on ResNet-101- $C_5$  (à trous, stride = 16)

	FPN	Align	Soft NMS	AP	AP <sub>75</sub>	AP <sub>S</sub>	AP <sub>M</sub>	AP <sub>L</sub>
Faster R-CNN				53.5	58.3	1.2	31.1	57.2
R-FCN				60.0	69.9	1.0	47.1	61.9
Ablations	✓			63.8	76.1	2.9	53.5	65.2
		✓		64.0	75.8	1.8	50.9	65.8
Our method	✓	✓		65.2	78.2	4.4	56.1	66.2
	✓	✓	✓	<b>66.3</b>	<b>80.5</b>	<b>4.4</b>	<b>56.9</b>	<b>67.4</b>

## 4.3 Detection Results

The detection main results are shown in Table 2. Our method achieves 5.2 points improvement in AP and 8.3 points in AP<sub>75</sub>. Moreover, our method improves the AP of small object by 4× compared to Faster R-CNN and R-FCN. The small object detection in our dataset is pretty difficult because of complex background and partial occlusion. One important reason is the use of FPN. FPN utilizes the feature maps of lower level, which have much larger size and keep many details.

We run some ablations to evaluate the components of our method. As shown in Table 2, our approach does benefit from advanced network like FPN, especially boosting the AP of small and medium objects. The finer multi-level feature maps address the multi-scale problem in real classrooms. Position-sensitive RoIAlign layer improve the accuracy our approach through reducing the misalignment in extracting feature for RoI (Fig. 3).

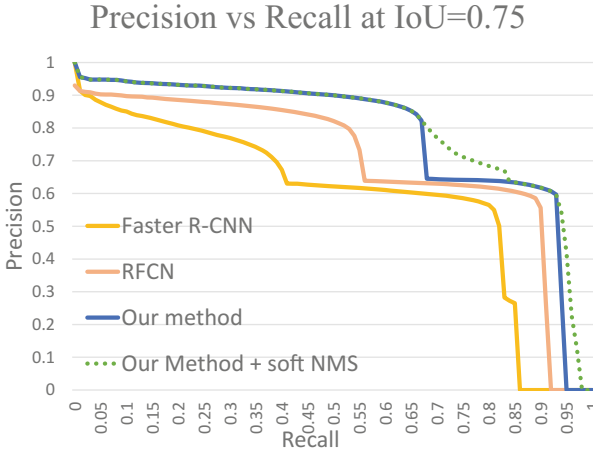


Fig. 3. Precision-recall curve on test dataset (IoU = 0.75)

## 5 Conclusion

We propose an improved method which incorporates R-FCN with FPN and utilizes position-sensitive RoIAlign layers. To address the multi-scale problem in real classrooms, we introduce FPN in R-FCN architecture to obtain multi-level feature maps. Meanwhile, position-sensitive RoIAlign layer is utilized to extract more accurate features. The experimental results show that our method achieves an impressive improvement in the multi-object detection of real classrooms.



**Acknowledgements.** The research was supported by NSFC (No. 61671290), the Key Program for International S&T Cooperation Project of China (No. 2016YFE0129500), and Shanghai Committee of Science and Technology (No. 17511101903).

## References

1. Bodla, N., Singh, B., Chellappa, R., Davis, L.S.: Soft-NMS—improving object detection with one line of code. In: 2017 IEEE International Conference on Computer Vision (ICCV), pp. 5562–5570. IEEE (2017)
2. Dai, J., Li, Y., He, K., Sun, J.: R-FCN: object detection via region-based fully convolutional networks. In: Advances in Neural Information Processing Systems, pp. 379–387 (2016)
3. Erhan, D., Szegedy, C., Toshev, A., Anguelov, D.: Scalable object detection using deep neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2147–2154 (2014)
4. Girshick, R.: Fast r-cnn. arXiv preprint [arXiv:1504.08083](https://arxiv.org/abs/1504.08083) (2015)
5. 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)
6. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN. In: 2017 IEEE International Conference on Computer Vision (ICCV), pp. 2980–2988. IEEE (2017)
7. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
8. Jaderberg, M., Simonyan, K., Zisserman, A., et al.: Spatial transformer networks. In: Advances in Neural Information Processing Systems, pp. 2017–2025 (2015)
9. 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)
10. LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L.D.: Backpropagation applied to handwritten zip code recognition. *Neural Comput.* **1**(4), 541–551 (1989)
11. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: CVPR, vol. 1, p. 4 (2017)
12. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431–3440 (2015)
13. Pont-Tuset, J., Arbelaez, P., Barron, J.T., Marques, F., Malik, J.: Multiscale combinatorial grouping for image segmentation and object proposal generation. *IEEE Trans. Pattern Anal. Mach. Intell.* **39**(1), 128–140 (2017)
14. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: Advances in Neural Information Processing Systems, pp. 91–99 (2015)
15. Ren, S., He, K., Girshick, R., Zhang, X., Sun, J.: Object detection networks on convolutional feature maps. *IEEE Trans. Pattern Anal. Mach. Intell.* **39**(7), 1476–1481 (2017)
16. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.* **115**(3), 211–252 (2015)
17. Uijlings, J.R., Van De Sande, K.E., Gevers, T., Smeulders, A.W.: Selective search for object recognition. *Int. J. Comput. Vis.* **104**(2), 154–171 (2013)
18. Zitnick, C.L., Dollár, P.: Edge boxes: locating object proposals from edges. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8693, pp. 391–405. Springer, Cham (2014). [https://doi.org/10.1007/978-3-319-10602-1\\_26](https://doi.org/10.1007/978-3-319-10602-1_26)



# Deep CRF-Graph Learning for Semantic Image Segmentation

Fuguang Ding , Zhenhua Wang  <sup>(✉)</sup>, Dongyan Guo, Shengyong Chen, Jianhua Zhang, and Zhanpeng Shao

College of Computer Science and Technology, Zhejiang University of Technology,  
Hangzhou, People's Republic of China  
zhhwang@zjut.edu.cn

**Abstract.** We show that conditional random fields (CRFs) with learned heterogeneous graphs outperforms its pre-designated homogeneous counterparts with heuristics. Without introducing any additional annotations, we utilize four deep convolutional neural networks (CNNs) to learn the connections of one pixel to its left, top, upper-left, upper-right neighbors. The results are then fused to obtain the super-pixel-level CRF graphs. The model parameters of CRFs are learned via minimizing the negative pseudo-log-likelihood of the potential function. Our results show that the learned graph delivers significantly better segmentation results than CRFs with pre-designated graphs, and achieves state-of-the-art performance when combining with CNN features.

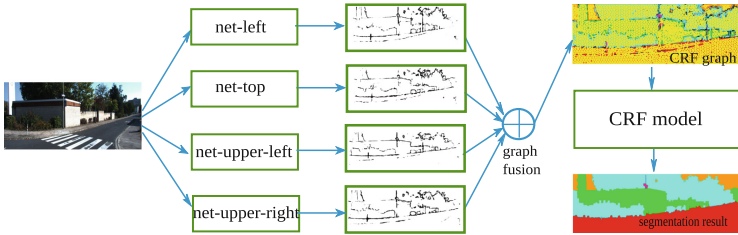
**Keywords:** Graph learning · Conditional random field · Segmentation

## 1 Introduction

Semantic image segmentation is an important task in pattern recognition. This problem has widely been explored in this field and numerous techniques have been presented, among which, CRF-based approaches play important roles. Indeed, segmentation with CRF significantly outperforms the approaches without using CRF, especially when the local feature representations are weak [4, 21]. We can group CRF-based approaches into two categories. The first category appends a CRF at the end of neural nets and trains all parameters (including both CRF parameters and network weights) jointly [14, 15, 17, 22, 25] via the well-known back-propagation algorithm. Here CRF is implemented as a specific fully-connected layer within the network, and embedding other structures (like grids or trees) under such framework is impossible. The second strategy is the so-called piecewise training which trains deep neural network and CRF separately [1, 4-7, 9, 19, 21].

Our approach is of the second category, among which, great effort has been put on learning effective features for segmentation with deep neural networks. While the problem of seeking proper topology of CRF graphs has not attracted much attention in this field. Rather, handcrafted ones like tree-structured graphs

[3] and complete graphs [4, 25] are typically used, since the former is easy to perform inference, and the latter can be seamlessly embedded into CNNs to achieve end-to-end training. Inspired by the success of deep learning on various challenging tasks, an interesting problem is can we learn proper CRF graphs directly from data with CNNs? This problem is challenging due to the fact that the solution space is too large, the graph structures are heterogeneous and the training data is extremely limited. To tackle this, we presents a deep learning framework which enables the learning of CRF graphs directly from training samples without using any additional annotations, and shows how segmentation benefits from learning CRF graphs. Specifically, we reduce the complex learning task to an easier problem: determining the connections of each pixel to its left, top, upper-left and upper-right neighbors, which are covered respectively by the four CNNs in Fig. 1. The outputs of the four CNNs are then fused to generate the final CRF graph with respect to an over-segmentation of the image (Sect. 2), which are taken to build the CRF model. We empirically verify that learned graphs outperform pre-defined adjacency and tree structures on two challenging segmentation tasks.



**Fig. 1.** We propose to learn CRF graphs with four convolutional neural networks, *i.e.* *net-left*, *net-top*, *net-upper-left*, *net-upper-right*. The learned graphs are fused to obtain the final graph in our CRF model.

## 2 Learn CRF Graphs

Liu *et al.* [16] leverage richer convolutional features (RCF) to detect object boundaries, where a neural network is taken to predict a probability for each pixel, representing its edginess. We observe that this detection problem shares a number of similar properties with the graph learning task in this paper, due to (1) both problems need to estimate a probability for each pixel, representing either its edginess for boundary detection, or the confidence that the pixel and its neighboring pixel in a particular direction are of the same object; (2) both problems rely on both local observation and global context to calculate probabilities. This observation motivates us taking the same CNN structure in [16] to learn CRF graphs by fine-tuning the network parameters on our own task.



**Fig. 2.** Generating graph groundtruth. From left to right, the first diagram shows a toy example ( $4 \times 4$  image), where we consider the connections (represented by the outgoing edges) of each pixel (e.g. the red one) to its *left*, *top*, *upper-left*, *upper-right* neighbors. The second is an input image. The next shows the pixel-labelling groundtruth, and the last illustrates graph groundtruth for the *left* connection, where brighter color indicates stronger connections while darker color suggests weaker connections. (Color figure online)

## 2.1 Obtain Groundtruth from Pixel Labelling

We decompose the determination of CRF graphs into predicting the connections of each pixel to its surrounding pixels. In other words, we neglect long-range connections. Specifically, for each pixel in an image we consider the connections to its neighbors in four directions, *i.e.* *left*, *top*, *upper-left*, *upper-right* (see Fig. 2 left). For each direction, we train a convolutional neural network to estimate the connection in that direction by fine-tuning the model pre-trained on BSDS500 for edge detection (the model is available online [16]). To train the CNN model for a particular direction, we generate graph groundtruth (a black-white image) by the following two steps: (1) We create a zero matrix sized the same as image. For each pixel we set the entry of the corresponding location in the matrix to 255 if the its label is identical to the pixel neighboring to it in the specific direction. The resulting binary image is visually similar as the edge-detection result, see Fig. 2. (2) We thicken the edges within the generated binary image. For each pixel in an edge, we set values of the two pixels, which are the nearest neighbors of the pixel in the specific direction, to (56,56,56) and (161,161,161) respectively. The thickened images are used as groundtruth (Fig. 2(b)) within our CRF-graph learning task.

## 2.2 Training

For the training of CNNs, the batch size is 1 and the global learning rate is set to  $10^{-6}$  and is divided by 10 after every 10k iterations. The momentum and weight decay are set to 0.9 and 0.0002 respectively. We stop training when the loss reaches a flat value. As a result, we obtain four networks which cover the connectivity of neighboring pixels in four directions including *left*, *top*, *upper-left*, *upper-right* (see the outputs of these nets in Fig. 1 for example). The output of each net gives the probabilities that the connections exists in the corresponding direction. Intuitively, the probability is high if the neighboring pixels are visually similar in appearance, and vice versa.

### 2.3 CRF-Graph Fusion

Given the outputs of the four networks, to reduce problem sizes, we first over-segment images to obtain super-pixels, which are then taken to build our CRF model (though all evaluations in Sect. 4 are performed in pixel level as typically done in literature). For each pair of super-pixels  $(k, l)$  that adjacent to each other, we consider all their neighboring pixels in 8 directions, representing by a set  $A_{k,l} = \{(i, j) | i \in k, j \in l, i \text{ is } j\text{'s 8-neighbor}\}$ . For each  $(i, j) \in A_{k,l}$ , let  $p_{i,j}$  denote the output of the corresponding net. We define

$$P_{k,l} = \frac{1}{|A_{k,l}|} \sum_{(u,v) \in A_{k,l}} p_{u,v}, \quad (1)$$

which evaluates of the connectivity of two neighboring super-pixels within each CRF graph. Let  $e_{u,v} \in \{0, 1\}$  represents if the edge  $(u, v)$  exists (when  $e_{u,v} = 1$ ) or not (when  $e_{u,v} = 0$ ). We get CRF graphs according to

$$e_{i,j} = \begin{cases} 1 & \text{if } P_{u,v} > \delta; \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

Here  $\delta$  is a threshold to be determined via cross-validation. In this way, we can fuse the outputs of four nets and obtain the CRF graph at the super-pixel level.

## 3 CRF Representation

Let  $\mathbf{z}$  denote an arbitrary image,  $\mathbf{x}$  denote the labelling of all super-pixels in an image. Let  $x_i \in \mathcal{X}$  be the label of the  $i$ -th super-pixel. Let  $G = (V, E)$  denote the learned graph with the approach introduced in Sect. 2, where  $V = \{1, \dots, n\}$  is the node set for  $n$  super-pixels, and  $E = \{(i, j) \subseteq V \times V\}$ .

Given an input  $\mathbf{z}$  and the graph structure  $E$ , the probabilistic distribution function (PDF) of  $\mathbf{x}$  is given by:

$$P(\mathbf{x} | \mathbf{z}, \mathbf{w}) = \frac{1}{N(\mathbf{z}, \mathbf{w})} \exp \left( \sum_{i \in V} -w_u \log p_i(x_i | \mathbf{z}) + \sum_{(i,j) \in E} w_{p_1} h_1(x_i, x_j, \mathbf{z}) + w_{p_2} h_2(x_i, x_j, \mathbf{z}) + \mathbf{w}_c^\top \mathbf{1}(x_i, x_j) \right), \quad (3)$$

where  $\mathbf{w} = [w_u, w_{p_1}, w_{p_2}, \mathbf{w}_c]$  is the parameter to be learned for this distribution,  $\mathbf{w}_c \in \mathbf{R}^l$  ( $l = |\mathcal{X}| \times (|\mathcal{X}| + 1)/2$ ),  $\mathbf{1}(x_i, x_j)$  is an indicator vector with respect to labels  $x_i, x_j$ , which takes 0 at all positions except for the one indexed by  $(x_i, x_j)$ .  $N(\mathbf{z}, \mathbf{w})$  is the so-called normalization constant.

There are four terms in Eq. (3). Within the first term,  $p_i(x_i)$  gives the possibility that the  $i$ -th super-pixel takes a label  $x_i$  observing image  $\mathbf{z}$ . To obtain such measurement, we train a deep neural network (see Sect. 4 for details) with a softmax output layer. Since the networks output pixelwise probabilities, we compute  $p_i(x_i)$  for a super-pixel  $i$  by averaging the probabilities of all pixels

within the super-pixel. Terms  $h_1(x_i, x_j, \mathbf{z})$ ,  $h_2(x_j, x_j, \mathbf{z})$  share the same form. The definition for  $h_1(x_i, x_j, \mathbf{z})$  is

$$h_1(x_i, x_j, \mathbf{z}) = \begin{cases} \exp(-\|\mathbf{c}_i - \mathbf{c}_j\|_2) & \text{if } x_i \neq x_j, \\ 1 - \exp(-\|\mathbf{c}_i - \mathbf{c}_j\|_2) & \text{otherwise.} \end{cases} \quad (4)$$

Above  $\mathbf{c}$  denotes the color vector (HSV space) of a super-pixel. For  $h_2$ , we use the same form with  $h_1$  but replacing  $\mathbf{c}$  by  $\mathbf{l}$ , which denotes the location vector (the centroid) of a super-pixel. These functions are typically utilized to smooth the segmentation since neighboring super-pixels are encouraged to take the same label if they are close in distance and appearance. The last term is a dot product between the parameter vector  $\mathbf{w}_c$  and the indicator vector  $\mathbf{1}(x_i, x_j)$ . With this term, we are able to learn the compatibility of the labelling of neighboring super-pixels, without knowing colors or locations of them. This term is useful because some labelling configurations (*e.g.* car-road) make more sense than others (car-water as an example).

We estimate all model parameters via pseudo-log-likelihood estimation [12], where seeking optimal labelling of  $\mathbf{x}$  (*i.e.* maximum a posteriori (MAP) estimation) is solved via alpha-beta expansion.

## 4 Experimental Results

We evaluate the proposed approach using two well-known datasets, one is KITTI and the other is VOC 2012. For each dataset, we present both quantitative (using the intersection over union, IoU criterion) and qualitative results. To show the strength of the CRF model with learned graphs, we compare our approach (*CRF with learned graphs (LG-CRF)*) against the following approaches (*CRF with pre-defined homogeneous graphs*):

- *Super-pixel alone (Sp-alone)*: First we run convolutional networks and obtain pixel-level segmentation results. Then we get the super-pixel level segmentation by voting labels of all pixels within each super-pixel. This approach does not use CRF.
- *Adjacent CRF (Adj-CRF)*: The same as our approach except that the graphs are determined according to the adjacency of super-pixels.
- *Minimum spanning tree CRF (MST-CRF)*: The same as our approach again except that its graphs are minimum spanning trees built according to the summation of color and location distances among super-pixels.

Though these approaches perform super-pixel-wise segmentation, we compute their IoU results at pixel-level. Besides, we also include recent pixel-wise segmentation results for comparison where is possible.

### 4.1 Results on KITTI

KITTI odometry dataset [20] includes 120 training images and 83 testing images of 7 categories (object, road, building, tree/bush, sign/pole, sky and grass/dirt).

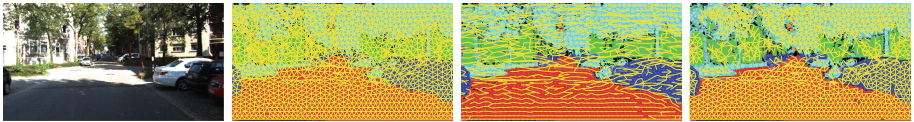


**Table 1.** Results (IoU) using KITTI odometry dataset.

Method	Object	Road	Building	Tree/bush	Sign/pole	Sky	Grass/dirt	Mean
Sem-Map [11]	70.39	92.11	76.68	64.35	1.880	89.30	25.23	52.91
Sp-alone	70.42	84.72	75.09	77.85	13.62	82.53	22.84	61.01
Adj-CRF	79.19	91.05	82.14	80.58	16.55	80.86	30.00	65.77
MST-CRF	75.48	90.20	82.53	79.68	<b>20.06</b>	89.08	27.96	66.43
LG-CRF	<b>79.37</b>	<b>92.65</b>	<b>84.46</b>	<b>82.92</b>	9.270	<b>91.32</b>	<b>32.29</b>	<b>67.47</b>

We fine-tune voc-fcn8s [18] and use the fine-tuned model to compute the pixel-wise segmentation. Then the results are further utilized to generate the *Sp-alone* segmentation and to compute  $p_i(x_i)$  in Eq. 3.

The results (with  $\delta = 0.9$ ) are provided in Table 1. Our approach (LG-CRF) gets the best result on all classes except for sign-pole. In general, LG-CRF improves around 1% over the second best (MST-CRF). Since *Sp-alone* can be viewed as a special CRF model with  $E = \emptyset$ , the only difference between *Sp-alone*, *Adj-CRF*, *MST-CRF* and *LG-CRF* is their graph structures. Hence the conclusions are: (1) The topology of CRF graphs is critical to the segmentation task; (2) CRFs with learned graphs (using deep CNN models) admits much better segmentation results compared with CRFs with hand-created graphs (*Adj-CRF* and *MST-CRF*). Figure 3 shows CRF graphs generated using different approaches. Clearly the learned graphs tend to connect super-pixels of the same category, while break connections across object boundaries. We also provide some qualitative results in Fig. 4.



**Fig. 3.** KITTI CRF graph structure. The first column shows a input image. The next, from left to right, show graphs (super-imposed on segmentation groundtruth) obtained using super-pixel adjacency, minimum spanning tree (based on color and location of super-pixels) and the deep learning approach introduced in Sect. 2, respectively. (Color figure online)



**Fig. 4.** Segmentation visualization of KITTI examples. The first column shows two inputs. The second column shows the groundtruth. The rest columns, from left to right, are results by Adj-CRF, MST-CRF and LG-CRF respectively.

Table 2. Results (IoU) on PASCAL VOC 2012 val set.

Method	bkg	Aero	Bike	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Table	Dog	Horse	mbike	Person	Plant	Sheep	Sofa	Train	Tv	Mean
DeepLab [4]	-	78.7	34.3	73.8	61.5	67.2	83.7	78.2	80.9	29.7	74.0	52.1	72.2	68.4	73.6	80.6	48.4	72.4	43.6	77.0	59.1	66.7
DPN [17]	-	84.8	37.5	80.7	66.3	67.5	84.2	76.4	81.5	33.8	65.8	50.4	76.8	67.1	74.9	81.1	48.3	75.9	41.8	76.6	60.4	67.8
FCRN+Bbs [23]	-	<b>88.3</b>	40.4	86.5	66.6	80.1	91.6	84.3	90.1	36.6	83.7	53.6	84.5	85.1	79.9	83.9	59.0	83.3	44.6	81.1	74.5	74.8
LMP Mean [2]	92.9	85.0	<b>79.6</b>	81.4	70.0	76.4	92.4	82.4	89.4	39.9	82.7	58.6	82.9	81.8	80.2	81.6	61.2	84.3	45.4	82.5	71.7	76.3
ResNet [10]	-	87.9	41.4	<b>89.5</b>	72.5	<b>80.7</b>	93.0	87.7	91.7	39.7	83.2	53.8	85.0	85.2	82.5	85.6	59.8	85.5	40.2	87.0	77.2	76.3
PDNs [24]	-	87.7	42.8	89.4	73.4	80.4	93.1	<b>88.8</b>	91.4	39.1	83.7	51.7	84.7	84.8	83.6	86.1	60.4	87.1	42.7	87.4	<b>78.2</b>	76.7
Sp-alone	95.2	88.2	59.8	86.8	74.1	80.0	93.0	85.6	92.2	40.1	90.6	67.6	89.9	88.3	<b>85.7</b>	<b>88.0</b>	65.4	90.8	55.6	87.7	74.6	80.0
Adj-CRF	94.8	86.4	47.6	85.2	72.2	79.9	93.3	85.1	92.1	40.4	89.8	<b>71.8</b>	89.3	87.4	84.3	86.1	62.7	89.8	<b>57.1</b>	86.6	73.9	78.8
MST-CRF	95.1	88.0	56.9	85.7	<b>74.6</b>	79.7	<b>93.4</b>	85.5	92.2	<b>41.3</b>	90.6	69.8	90.1	88.0	85.3	87.6	<b>65.9</b>	90.5	55.9	87.3	73.1	79.8
LG-CRF	<b>95.3</b>	88.2	60.0	86.4	74.1	80.4	93.2	85.9	<b>92.6</b>	40.7	<b>90.9</b>	69.0	<b>90.4</b>	<b>88.5</b>	85.5	<b>88.0</b>	65.8	<b>90.8</b>	55.9	<b>88.2</b>	74.8	<b>80.2</b>

## 4.2 Results on PASCAL VOC 2012

Pascal voc 2012 [8] is a benchmark for semantic segmentation which includes 20 object categories and one background class. This dataset is split into three subsets for training, validation and testing with 1,464, 1,449 and 1,456 images respectively. We use the RefineNet-Res101 model provided by Lin *et al.* [13] to get the pixel-wise segmentation. Afterwards the results are used to generate the *Sp-alone* segmentation and to compute  $p_i(x_i)$  in Eq. 3.

We present results on the validation set in Table 2 (with  $\delta = 0.9$ ). Our approach (LG-CRF) has marginal improvement compared with Sp-alone, Adj-CRF and MST-CRF. Specifically, LG-CRF performs best on 8 out of 21 categories (including background), which verifies the strength of the proposed approach. Interestingly, CRFs with hand-crafted graphs (*i.e.* Adj-CRF and MST-CRF) are slightly worse than Sp-alone, which is equivalent to a CRF with isolate graph (*i.e.*  $E = \emptyset$ ). It might be because the result of RefineNet-Res101 is excellent and CRF refining with inappropriate graphs deteriorate the segmentation.

## 5 Conclusion

We found that CRF graphs are of great importance to train proper CRF models for semantic segmentation. By reducing the complex graph-structure-learning problem to determining the connections of adjacent pixels, we can learn the topology CRF graphs with CNNs in a supervised manner without introducing additional annotations. With learned graphs, CRFs can be much more effective than CRFs with hand-crafted graphs. Indeed, the proposed approach outperforms baselines (*i.e.* CRFs with hand-crafted graphs) by large margins, and is competitive when comparing with the state-of-the-arts.

**Acknowledgement.** This research was partly supported by the Zhejiang Provincial Natural Science Foundation of China (LQ16F030007 and LQ18F030013), and by National Natural Science Foundation of China (U1509207, 61305021 and 61603341).

## References

1. Bell, S., Upchurch, P., Snavely, N., Bala, K.: Material recognition in the wild with the materials in context database. In: CVPR, pp. 3479–3487 (2015)
2. Buló, S.R., Neuhold, G., Kotschieder, P.: Loss max-pooling for semantic image segmentation. arXiv preprint [arXiv:1704.02966](https://arxiv.org/abs/1704.02966) (2017)
3. Cadena, C., Košecká, J.: Semantic segmentation with heterogeneous sensor coverages. In: ICRA, pp. 2639–2645. IEEE (2014)
4. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Semantic image segmentation with deep convolutional nets and fully connected crfs. arXiv preprint [arXiv:1412.7062](https://arxiv.org/abs/1412.7062) (2014)
5. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. arXiv preprint [arXiv:1606.00915](https://arxiv.org/abs/1606.00915) (2016)

6. Cogswell, M., Lin, X., Purushwalkam, S., Batra, D.: Combining the best of graphical models and convnets for semantic segmentation. arXiv preprint [arXiv:1412.4313](https://arxiv.org/abs/1412.4313) (2014)
7. Dai, J., He, K., Sun, J.: Boxesup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In: ICCV, pp. 1635–1643 (2015)
8. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (VOC) challenge. IJCV **88**(2), 303–338 (2010)
9. Farabet, C., Couprie, C., Najman, L., LeCun, Y.: Learning hierarchical features for scene labeling. TPAMI **35**(8), 1915–1929 (2013)
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR, pp. 770–778 (2016)
11. Kochanov, D., Ošep, A., Stücker, J., Leibe, B.: Scene flow propagation for semantic mapping and object discovery in dynamic street scenes. In: IROS, pp. 1785–1792. IEEE (2016)
12. Korč, F., Förstner, W.: Approximate parameter learning in conditional random fields: an empirical investigation. In: Rigoll, G. (ed.) DAGM 2008. LNCS, vol. 5096, pp. 11–20. Springer, Heidelberg (2008). [https://doi.org/10.1007/978-3-540-69321-5\\_2](https://doi.org/10.1007/978-3-540-69321-5_2)
13. Lin, G., Milan, A., Shen, C., Reid, I.: Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. arXiv preprint [arXiv:1611.06612](https://arxiv.org/abs/1611.06612) (2016)
14. Lin, G., Shen, C., van den Hengel, A., Reid, I.: Efficient piecewise training of deep structured models for semantic segmentation. In: CVPR, pp. 3194–3203 (2016)
15. Lin, G., Shen, C., Van Den Hengel, A., Reid, I.: Exploring context with deep structured models for semantic segmentation. TPAMI **40**(6), 1352–1366 (2018)
16. Liu, Y., Cheng, M.M., Hu, X., Wang, K., Bai, X.: Richer convolutional features for edge detection. In: CVPR, pp. 5872–5881. IEEE (2017)
17. Liu, Z., Li, X., Luo, P., Loy, C.C., Tang, X.: Semantic image segmentation via deep parsing network. In: ICCV, pp. 1377–1385 (2015)
18. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: CVPR, pp. 3431–3440 (2015)
19. Noh, H., Hong, S., Han, B.: Learning deconvolution network for semantic segmentation. In: ICCV, pp. 1520–1528 (2015)
20. Ošep, A., Hermans, A., Engelmann, F., Klostermann, D., Mathias, M., Leibe, B.: Multi-scale object candidates for generic object tracking in street scenes. In: ICRA, pp. 3180–3187. IEEE (2016)
21. Papandreou, G., Chen, L.C., Murphy, K., Yuille, A.L.: Weakly-and semi-supervised learning of a DCNN for semantic image segmentation. arXiv preprint [arXiv:1502.02734](https://arxiv.org/abs/1502.02734) (2015)
22. Schwing, A.G., Urtasun, R.: Fully connected deep structured networks. arXiv preprint [arXiv:1503.02351](https://arxiv.org/abs/1503.02351) (2015)
23. Wu, Z., Shen, C., Hengel, A.V.D.: High-performance semantic segmentation using very deep fully convolutional networks. arXiv preprint [arXiv:1604.04339](https://arxiv.org/abs/1604.04339) (2016)
24. Zhang, R., Yang, W., Peng, Z., Wang, X., Lin, L.: Progressively diffused networks for semantic image segmentation. arXiv preprint [arXiv:1702.05839](https://arxiv.org/abs/1702.05839) (2017)
25. Zheng, S., Jayasumana, S., Romera-Paredes, B., Vineet, V., Su, Z., Du, D., Huang, C., Torr, P.H.: Conditional random fields as recurrent neural networks. In: ICCV, pp. 1529–1537 (2015)



# Unrest News Amount Prediction with Context-Aware Attention LSTM

Xiuling Wang<sup>1</sup>, Hao Chen<sup>1</sup>, Zhoujun Li<sup>1(✉)</sup>, and Zhonghua Zhao<sup>2</sup>

<sup>1</sup> Beihang University, Beijing, China  
lizj@buaa.edu.cn

<sup>2</sup> National Computer Network Emergency Response Technical Team,  
Coordination Center of China, Beijing, China

**Abstract.** Accurately predicting social unrest events is crucial to improve public security. Currently, with the large scale news event datasets available such as GDELT, we can use the amount of unrest news to estimate the risk of instability which is particularly helpful in resource allocation and policy making. Thus in this paper we propose a context-aware attention based long short-term memory (LSTM) prediction framework named CA-LSTM to accurately predict the amount of unrest news of each country or state in the future. Specifically, we first use LSTM to learn the hidden representation from the raw time series data, and then we employ a temporal attention mechanism to learn the importance weight of each time slot. Finally, a fully connected layer is adopted to predict the future unrest news amount by combining the context information and the time series embedding vectors. We conduct extensive experiments on the GDELT data of the United States, and the results demonstrate the effectiveness of the proposed framework.

**Keywords:** Unrest event prediction · Attention LSTM  
Context-aware

## 1 Introduction

Social event prediction, which aims to mine the causal relationship between the previously-observed event data and their future tendency has been widely explored in unrest prediction, such as civil wars [1], disease outbreaks [2], crimes [3] and so on. Social unrests, also known as civil disorder, usually cause considerable economic losses. Thus the accurate prediction of unrests is important in protecting public security. In recent years, social unrest event prediction has drawn remarkable attention by both research communities and governments [4, 5].

Current studies on social unrest prediction can be categorized into the following three categories. (1) Traditional machine learning based methods: researchers always manually extract features and train a machine learning model such as LASSO, SVM and random forest. Korkmaz [6] used LASSO to predict the probability of the occurrence of the civil unrest events. (2) Time series analysis based

methods: this type of methods focus on applying statistical time series techniques. Yonamine [7] employed ARFIMA model to predict the material conflict for Afghanistan districts. (3) Neural network based methods: neural networks especially LSTMs have shown their success in sequential-based learning tasks. Smith [8] showed that LSTM achieved a better performance in predicting the material conflict amount on the GDELT data. However, due to the unexplainable and ambiguous characteristics of time series data, the performance of the above existing methods is largely limited in unrest event news amount prediction.

Accurately predicting the unrest events faces the following three major challenges. First, time series data, unlike pictures, are hard to comprehend intuitively. Second, different data points of the historical time series data have different contribution to the prediction of their future trend. Thus it is challenging for traditional shallow models to learn the weights of the historical time series in different time intervals. Finally, unrest news prediction is influenced by both tendencies and near days value, which is difficult to trade off.

To address the above challenges, we propose a context-aware attention based long short-term memory prediction framework named CA-LSTM to accurately predict the amount of unrest event news. Our model has three parts: the LSTM encoder, the attention layer and the context-aware fully connected layer. Specifically, we first use a LSTM to get the hidden representation of the input sequence, which helps us to capture the underlying relationship of raw data. Then we apply an attention mechanism to automatically learn the weights of the hidden representations. Finally a context-aware fully connected layer is used to combine near historical target data and the weighted representation vectors.

Our main contributions can be summarized as follows: (a) We propose a novel context-aware attention-based long short-term memory model, which can extract the trends information automatically and predict the future amount with the help of the context information. (b) The proposed model is evaluated on the GDELT data of US states, and the results demonstrate its superior performance compared to baseline models.

## 2 GDELT Dataset

GDELT (Global Data on Events, Location, and Tone) [9] is the largest open real-time event database of human society, which monitors the world’s broadcast, print, and web news of nearly every corner of every country [10]. GDELT project has several databases, here we focus on GDELT 1.0 event database. In this dataset, GDELT extracts several events from each news report, and categorizes the events into 4 classes and 20 categories. The 4 classes consist of Verbal Cooperation, Material Cooperation, Verbal Conflict and Material Conflict. Each class is made up of 5 categories separately. We count the number of the events of each class and category for each country or state as its news amount information. The event Material Conflict captures unrest information, which can be seen as the indication of political instability. Thus, our goal is to predict the amount of material conflict with the historical amount of twenty categories event. As there

are varieties of noise in the raw data for news production, we adopt moving average method to extract much more meaningful information from the raw series. Small smooth window size can extract the high frequency, while big smooth window size can get low frequency information. Besides, we use the Min-Max normalization method to scale the data into the range (0, 1).

### 3 Notation and Problem Statement

In this section, we first introduce the mathematical notations used in this work and formally define the studied problem.

We use  $\{t_1, t_2, \dots, t_K\}$  to denote the time slots of the training dataset, where  $K$  is the number of the time slots.  $X_k^m$  denotes the news amount of category  $m$  at time slot  $t_k$ .  $X_k = \{X_k^1, X_k^2, \dots, X_k^M\} \in \mathbf{R}^M$  denotes the news amount of all categories from 1 to  $M$  in the time slot  $t_k$ , where  $M$  is the total categories of news types.  $y_k \in \mathbf{R}$  denotes the amount of unrest event news in the time slot  $t_k$ .

The studied problem can be defined as: Given the raw news amount time series data of a country or region  $\{X_t|t = 1, \dots, T\} \in \mathbf{R}^{T \times M}$ , we aim to predict the amount of unrest news  $y_{T+1}$  of the next time slot  $t_{T+1}$ .

### 4 Model

The overall architecture of the context-aware attention-based long short-term memory network (CA-LSTM) is shown in Fig. 1. It consists of the following three parts: the LSTM encoder, the attention layer and the context-aware fully connected layer. Next we will introduce these three parts in detail.

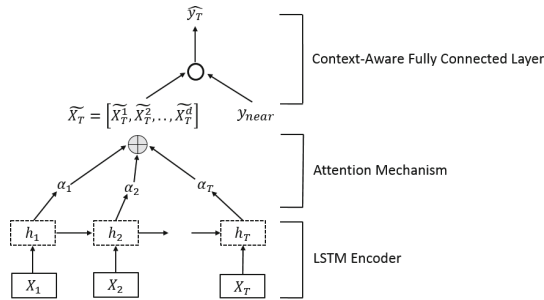


Fig. 1. Framework of the proposed CA-LSTM framework

#### 4.1 LSTM Encoder

Given the series of event news amount  $X = (X_1, \dots, X_T)$  with  $X_t \in \mathbf{R}^M$ , we use a LSTM unit as a mapping function to capture the latent representation of the time series data.

Each LSTM unit has a memory cell, which is controlled by three gates: the forget gate, the input gate and the output gate. At time step  $t$ , there are three inputs: the current input  $X_t$ , the previous hidden state  $h_{t-1}$  and the current cell state  $C_t$ . The gates  $f_t$ ,  $i_t$  decide how much information in previous time intervals is forgotten and how much new information is added to the current cell state respectively.

$$f_t = \sigma(W_f \cdot [h_{t-1}, X_t] + b_f) \quad i_t = \sigma(W_i \cdot [h_{t-1}, X_t] + b_i) \quad (1)$$

$$\tilde{C}_t = \tanh(W_c \cdot [h_{t-1}, X_t] + b_c) \quad C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t \quad (2)$$

where  $\sigma$  denotes a sigmoid function,  $C_t$  is the new cell state at time  $t$ ,  $[h_{t-1}, X_t] \in \mathbf{R}^{d+M}$ , and  $W_f, W_i, W_c \in \mathbf{R}^{d \times (d+M)}$ ,  $b_f, b_i, b_c \in \mathbf{R}^d$ .

Finally the hidden state  $h_t$  is controlled by the cell state  $C_t$  and the output gate  $o_t$ .

$$o_t = \sigma(W_o \cdot [h_{t-1}, X_t] + b_o) \quad h_t = o_t \cdot \tanh(C_t) \quad (3)$$

where  $W_o \in \mathbf{R}^{d \times (d+M)}$ ,  $b_o \in \mathbf{R}^d$ .

Based on such an architecture, we can get the hidden representation  $(h_1, \dots, h_T)$ .

#### 4.2 Attention Layer

The news amount of different previous time intervals may have different influences to the prediction of the news amount in the future. To take this into consideration, we introduce the attention mechanism to automatically mine the data points which are significant to prediction and weighted sum the representation of the hidden states to obtain a reasonable representation vector of the historical time series data. Specifically, given the  $T$ -th series  $(h_1, \dots, h_T) \in \mathbf{R}^{T \times d}$

$$u_t^i = \tanh(W_u \cdot h_t^i + b) \quad \alpha_t^i = \frac{\exp(u_t^i)}{\sum_t u_t^i} \quad \tilde{X}^i = \sum_t \alpha_t^i \cdot h_t^i \quad (4)$$

where  $W_u \in \mathbf{R}^{T \times d}$ ,  $\tilde{X}^i \in \mathbf{R}^d$ . The attention weight  $\alpha_t^i$  represents the importance of the  $t$ -th hidden state with the  $i$ -th dimension for prediction.  $\tilde{X}^i$  aggregates all the weighted vectors across the  $T$  time steps. The output can be seen as a high level representation of the raw time series data.

#### 4.3 Context-Aware Layer

The final prediction consider both tendencies information and the nearby values. Therefore, we design a context-aware fully connected layer to combine the representation vectors  $\tilde{X} = (\tilde{X}^1, \dots, \tilde{X}^d) \in \mathbf{R}^d$  and nearby values of unrest news



amount  $y_{past} = (y_{T-T'}, \dots, y_T) \in \mathbf{R}^{T'}$ . The observation of previous outputs  $y_{past}$  contains the context information.

$$\hat{y}_{T+1} = \sigma(\tilde{W} \cdot [y_{past}, \tilde{X}] + \tilde{b}) \quad (5)$$

$\sigma(x)$  denotes sigmoid function,  $\tilde{W}$  denotes the parameter vector,  $\tilde{W} \in \mathbf{R}^{T'+d}$ ,  $\tilde{b} \in \mathbf{R}$ .

**Model Training.** We use batch stochastic gradient descent (SGD) to train the model. Each region corresponds to a batch. The proposed CA-LSTM is smooth and differentiable, so the parameters can be learned by standard back propagation with squared loss as the objective function:

$$L(y_i, \hat{y}_i) = \sum (y_i - \hat{y}_i)^2 \quad (6)$$

## 5 Experiments

### 5.1 Experimental Setting

We evaluate our model on the GDELT data, which is collected from Jan 1, 2012 to Jan 11, 2018 in 50 states of US. For each state, there are total 2202 time slots. The input features contain the amount sequence of 20 categories and historical amount of unrest events  $y_{past}$ . We train the CA-LSTM framework based on  $T = 60$  time slots and past  $T' = 7$  days target values. Our aim is to predict the news amount of the material conflict event in the next day. We use the first 80% time slots for training, and remaining 20% time slots for testing. RMSE (Root Mean Square Error) and MAE (Mean Average Error) are adopted as the evaluation metrics, which is defined as follows:

$$MSE = \sqrt{\frac{1}{N} \frac{1}{K-T} \sum_{n=1}^N \sum_{k=T}^K (y_n^k - \hat{y}_n^k)^2} \quad MAE = \frac{1}{N} \frac{1}{K-T} \sum_{n=1}^N \sum_{k=T}^K |y_n^k - \hat{y}_n^k| \quad (7)$$

where  $N$  is the number of states,  $K$  is the length of time slots in each state,  $T$  is the windows size of the input features,  $\hat{y}_n^k$  denotes the predicted amount of time slot  $k$  and state  $n$ ,  $y_n^k$  denotes the ground truth amount of it.

### 5.2 Results

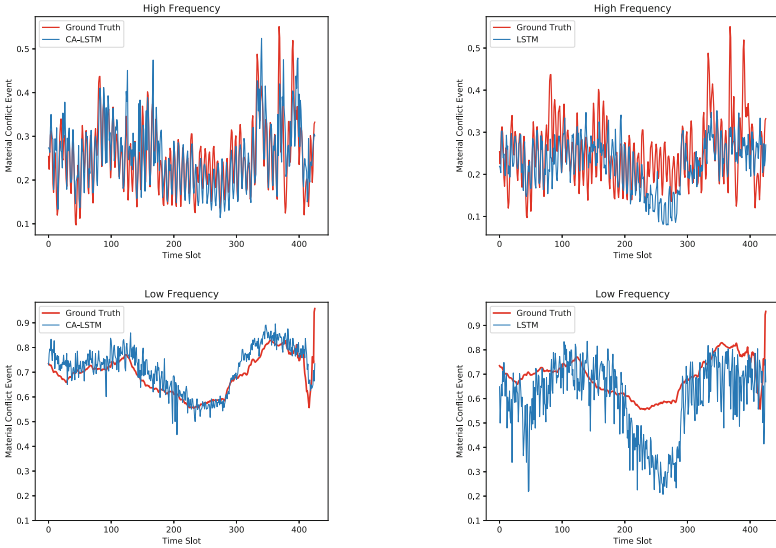
As mentioned in Sect. 2, the average moving method is used to smooth out the noise of the random outliers. In order to study the effect of the high and low frequency information on the model performance, we set the smoothing window size to 3 and 99 respectively. Our model is based on non-textual features, therefore we select LR, ARMA, LASSO and LSTM as baseline models. The experimental results are shown in Table 1. From this table, we can see that: LR achieves the worst performance, which indicates LR cannot handle with complex

**Table 1.** Overall performance comparison

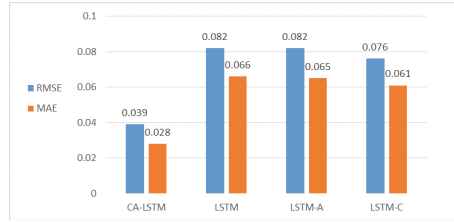
	High frequency		Low frequency	
	RMSE( $\times 10^{-1}$ )	MAE( $\times 10^{-1}$ )	RMSE( $\times 10^{-1}$ )	MAE( $\times 10^{-1}$ )
LR	1.51	1.18	1.38	1.07
ARMA	1.25	0.83	1.25	0.83
LASSO	0.95	0.75	0.96	0.60
LSTM	0.82	0.66	0.94	0.60
CA-LSTM	0.39	0.28	0.41	0.24

news amount series data in our studied problem. LASSO outperforms ARMA, as ARMA only considers near values  $(y_1, \dots, y_T)$  and ignores the other input features  $(X_1, \dots, X_T)$ . Compared with LR, LASSO achieves a lower loss because LASSO uses regularization to select helpful features. CA-LSTM performs much better than LSTM, as LSTM don't consider the different contribution of each time slot and ignores context information. The loss of CA-LSTM model is significantly lower than other four models, which indicates the strength of the attention LSTM layer and the context information.

**Case Study.** Here we shows the real prediction in California, US. Figure 2 depict the prediction results on the amount of unrest news in the next day by using CA-LSTM and LSTM. One can see the prediction results of CA-LSTM in both cases can more accurately fit the ground truth curves, which demonstrates the effectiveness of CA-LSTM.



**Fig. 2.** Prediction result of the high frequency and low frequency data



**Fig. 3.** Impact of the components on the model performance

**Evaluation on Model Components.** To evaluate whether each component of CA-LSTM can contribute to a better performance, we compare CA-LSTM with original LSTM, LSTM-A: LSTM unit with attention mechanism and LSTM-C: LSTM layer with context information. The experimental results are shown in Fig. 3. From this figure, one can see that LSTM-A achieves the worst performance, which indicates the importance of the context information. Moreover, CA-LSTM outperforms all the other variations which verifies that the context vector and the embedded vectors are complementary rather than conflicting. Using attention layer or context information individually cannot achieve a satisfactory performance.

## 6 Related Work

Recently, an increasing number of researches focus on social network [11, 12]. The analysis of social unrest events can be categorized into two main types: event detection and event prediction. Next we will review related works along the two research lines.

**Event Detection:** There is a large amount of work on the detection and identification of the various ongoing events, including disease outbreaks defection [13], earthquakes detection [14] and various other types of events detection [15]. However, instead of forecasting events in the future, these approaches typically uncover them only after they have occurred.

**Event Prediction:** Event prediction has also been explored in a variety of applications, including urban traffic [16], disease outbreaks prediction [2], social unrest event prediction [1], and crimes prediction [3]. Most recent social unrest event prediction techniques can be categorized into three types: planned event forecasting, machine learning based prediction, and time series mining based prediction.

## 7 Conclusion

In this paper, we proposed a novel CA-LSTM model in unrest event amount prediction. CA-LSTM can effectively extract the trend information with attention LSTM and make accurately prediction by combing the tendencies and the

context information. Evaluations on the GDELT data shows the strength of CA-LSTM in the prediction of unrest event amount in realistic world. And the comparison between CA-LSTM and LSTM proves the necessary of the context information which might be ignored. CA-LSTM is a much stronger baseline in the prediction of sequential data. Researchers focus on similar time series prediction problems can be inspired by our work. In the future, we are going to employ CA-LSTM to predict other event class.

**Acknowledgments.** This work was supported in part by the Natural Science Foundation of China (Grand Nos. U1636211, 61672081, 61370126), and Beijing Advanced Innovation Center for Imaging Technology (No. BAICIT-2016001) and National Key R&D Program of China (No. 2016QY04W0802).

## References

1. Qiao, F., et al.: Predicting social unrest events with hidden markov models using GDELT. *Discrete Dyn. Nat. Soc.* **2017**, 1–13 (2017)
2. Achrekar, H., Gandhe, A., Lazarus, R., Yu, S.H., Liu, B.: Predicting flu trends using twitter data. In: *Computer Communications Workshops*, pp. 702–707 (2011)
3. Zhao, X., Tang, J.: Modeling temporal-spatial correlations for crime prediction. In: *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, CIKM 2017*, pp. 497–506. ACM, New York (2017). <http://doi.acm.org/10.1145/3132847.3133024>
4. Stoll, R.J., Subramanian, D.: Hubs, authorities, and networks: predicting conflict using events data (2006)
5. Weidmann, N.B., Ward, M.D.: Predicting conflict in space and time. *J. Confl. Resolut.* **54**(6), 883–901 (2010)
6. Korkmaz, G., et al.: Combining heterogeneous data sources for civil unrest forecasting, pp. 258–265 (2015)
7. Yonamine, J.E.: Predicting future levels of violence in afghanistan districts using gdelt. Unpublished Manuscript (2013)
8. Smith, E.M., Smith, J., Legg, P., Francis, S.: Predicting the occurrence of world news events using recurrent neural networks and auto-regressive moving average models. In: Chao, F., Schockaert, S., Zhang, Q. (eds.) *UKCI 2017. AISC*, vol. 650, pp. 191–202. Springer, Cham (2018). [https://doi.org/10.1007/978-3-319-66939-7\\_16](https://doi.org/10.1007/978-3-319-66939-7_16)
9. GDELT: Gdelt. <https://www.gdeltproject.org/>
10. Leetaru, K., Schrodtt, P.A.: Gdelt: Global data on events, location and tone, 1979–2012. In: *Annual Meeting of the International Studies Association* (2013)
11. Wang, S., Hu, X., Yu, P.S., Li, Z.: MMRate: inferring multi-aspect diffusion networks with multi-pattern cascades, pp. 1246–1255 (2014)
12. Zhan, Q., Zhang, J., Wang, S., Yu, P.S., Xie, J.: Influence maximization across partially aligned heterogenous social networks. In: Cao, T., et al. (eds.) *PAKDD 2015, Part I. LNCS (LNAI)*, vol. 9077, pp. 58–69. Springer, Cham (2015). [https://doi.org/10.1007/978-3-319-18038-0\\_5](https://doi.org/10.1007/978-3-319-18038-0_5)
13. Signorini, A., Segre, A.M., Polgreen, P.M.: The use of twitter to track levels of disease activity and public concern in the U.S. during the influenza a h1n1 pandemic. *Plos One* **6**(5), e19467 (2011)

14. Sakaki, T., Okazaki, M., Matsuo, Y.: Earthquake shakes twitter users: real-time event detection by social sensors. In: Proceedings of the International World Wide Web Conference, pp. 851–860 (2010)
15. Jin, F., et al.: Misinformation propagation in the age of twitter. *Computer* **47**(12), 90–94 (2014)
16. Wang, S., et al.: Estimating urban traffic congestions with multi-sourced data. In: IEEE International Conference on Mobile Data Management, pp. 82–91 (2016)



# Image Captioning with Relational Knowledge

Huan Yang, Dandan Song<sup>(✉)</sup>, and Lejian Liao

Beijing Engineering Research Center of High Volume Language Information Processing and Cloud Computing Applications, Beijing Key Laboratory of Intelligent Information Technology, School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, China  
{yh,sdd,liaolj}@bit.edu.cn

**Abstract.** People have learned extensive relational knowledge from daily life. This is one of the facts that enables human to describe the information from images easily. In this paper, we propose a novel framework called Image Captioning with Relational Knowledge (*ICRK*) that combines relational knowledge with image captioning model and utilizes relational knowledge to strengthen the learning process of representing words. As more precise syntactic and semantic word relationships were learned, the image captioning model acquires more semantic features that help to generate more accurate image descriptions. Experiments on several benchmark datasets, using automatic evaluation metrics, have all demonstrated that our model can significantly improve the quality of image captioning.

**Keywords:** Image captioning · Relational knowledge  
Word embedding

## 1 Introduction

Image captioning, a challenging task which combines Natural Language Processing with Computer Vision, has attracted more and more attention recently. Generating the descriptions of images automatically not only is of benefit for applications like image retrieval but also helps visually impaired people to see the world. It is so important that has been treated as a core problem in Computer Vision.

Recognizing and describing details in images is natural and easy for people. However, it can be a challenging task for image captioning models. One of the important reasons is that when looking at an image, people are not just recognizing a large number of objects in it, but also able to detect the relationship between them. For example, when we see “*girl*” and “*bed*” in an image, we will naturally describe it as “*A girl is sitting on the bed*”, and when given an object “*meal*” and the relationship “*is presented in*”, we can also easily come up with “*tray*” as the object where the meal is presented in. Because of the relational

knowledge people have, we can recognize the objects in the image more accurate and make a description more fluent. In contrast, the image captioning model cannot do it without learning this relational knowledge.

Continuous skip-gram model and continuous bag-of-words model (CBOW) and [12] have been proposed for computing continuous vector representations of words from very large data sets, and it has been proven that these models can learn high-quality word vectors from huge data. However, these models learned word representations from the continuously distributed representation of the context, so if there are little context information about two syntactically or semantically similar words, they cannot learn the relationship between them. In that case, when we put these word representations into image captioning model, the model that has learned relational knowledge will perform better than the model that hasn't learned. Furthermore, learning from the amount of context could be noisy or biased, and these word representations cannot reflect the inherent relationship between words.

In order to combine relational knowledge with image captioning model and get better word representations, we propose a novel model that incorporates the relational knowledge of words from knowledge graph into the learning process and treats relational knowledge as regularization function. Concretely, the main contribution of this paper is proposing a new image captioning algorithm which combines relational knowledge, and we define a new learning objective to strengthen the learning of word representation in image captions. We validate the effectiveness of this approach on several datasets in which we outperform competing methods and achieve state-of-the-art consistently across different evaluation metrics.

## 2 Related Work

Image captioning model can be divided into two categories generally: bottom-up and top-down. Bottom-up approaches use the visual concepts, objects detected from the image and pretrained neural network to get the words corresponding to these visual features, and then combine these words into sentences using language models. Representative works include [5, 8], and these methods rely on the effectiveness of the visual detectors and the ability of language model to generate sentences. However, unlike bottom-up approaches need to detect visual concept, words and put them together, top-down approaches can be trained from end to end. These approaches [6, 11, 16, 20] use a Convolutional Neural Network (CNN) to extract image features and combine these features with Recurrent Neural Network (RNN) to accomplish image captioning. The main difference between these approaches is that different methods use different CNN and RNN.

We notice that word representations are vital for image captioning no matter what model we use as the description of an image is organized by single words. Some recent effort, such as continuous skip-gram model and CBOW model [12], have attempted to learn word representations that can capture both the syntactic and the semantic information among words. However, in prior work [6], little change has been found in final performance of image captioning when adding these trained word vectors. In contrast, inspired by a popular study on the multi-relation model [2] that builds relationships between entities, we observe that there are also relationships between the objects in the image and this feature can be used in image captions. Instead of putting the word vectors which trained by word2vec [12] model into the image captioning model directly, we extract relational knowledge from the descriptions and extend the objective function of word2vec model by combining the relational knowledge as regularization function. What's more, instead of using the popular knowledge graphs, such as Freebase [1] and WordNet [14], to train our model, we build a knowledge base by our own which is tailored to this task without much noisy.

### 3 Proposed Model

#### 3.1 Overall Framework

Following several previous works [6, 11, 16, 20], we use a CNN to extract image features and RNN to connect images features with sentences features. In this work, our particular design is combining relational knowledge with image captioning model. We extract relational triplets from the descriptions of images and use this relational knowledge to construct semantic features, and then combine these semantic features with visual features of images to generate the descriptions of images automatically. Our overall image captioning model is illustrated in Fig. 1. We describe our method to construct semantic features based on relational knowledge in Sect. 3.2. In Sect. 3.3 we outline the architecture of our image captioning model.

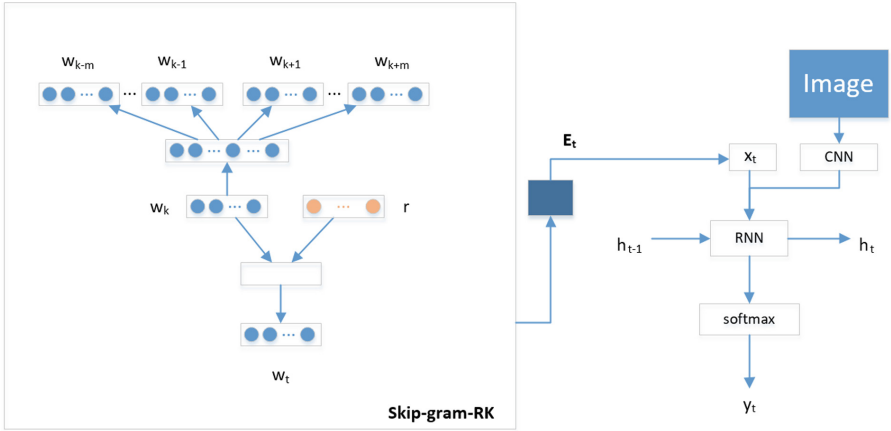
#### 3.2 Relational Knowledge with Word Representations

We adopt the continuous skip-gram model as the basis of the proposed relational knowledge embedding framework<sup>1</sup>. It is a word embedding model using a neural network architecture and has been proved efficient for learning high-quality distributed vector representations. The continuous skip-gram model focuses on finding word representations that are meaningful for predicting the surrounding words in a sentence.

---

<sup>1</sup> Note that although we use the continuous skip-gram model as an example to illustrate our framework, the similar framework can be developed on the basis of any other word embedding models.





**Fig. 1.** An illustration of our image captioning model ICRK. It is comprised of a CNN, a RNN and our Skip-gram-RK model.

Given a sequence of training words  $w_1, w_2, w_3, \dots, w_K$ , the objective of the continuous skip-gram model is to maximize the log probability:

$$\xi = \sum_{w_k \in C} \log p(\text{Context}(w_k) | w_k) \quad (1)$$

where  $C$  are words in the vocabulary,  $\text{Context}(w_k)$  is the training context  $\{w_k - m, \dots, w_k - 1, w_k + 1, \dots, w_k + m\}$ , and  $m$  indicates the context window size to be  $2m + 1$ .

Since the relational knowledge in knowledge graph is usually represented in the triplet (*head, relation, tail*) (denoted  $(h, r, t)$ ), each of which often can be extracted from text. The principle of previously developed translation-based model [2] is that  $h + r \approx t$ , if  $(h, r, t)$  holds, the embedding of the head entity  $h$  plus the embedding of the relationship  $r$  should be close to the tail entity  $t$ , otherwise  $h + r$  should be far away from  $t$ .

Similarly to this approach, we extract the triplet  $(w_h, r, w_t)$  from training data, and it consists of two words  $w_h, w_t$  and the relationship  $r$  contacting them. To combine the relational knowledge with word embedding model, we assume that relationships between words can be interpreted as translation operations and they can be represented by vectors. The basic idea of our model is that  $w_h + r \approx w_t$ . However, instead of learning vectors embedding by minimizing a margin-based ranking criterion over the training set which results in complex combined optimization problem [19], we adopt an objective to maximize the probability as below:

$$J = \sum_{r \in R_{w_h}} \log p(w_t | w_h + r) \quad (2)$$

To incorporate relational knowledge into word representations learning system, we get the following combined objective  $D$ :

$$D = \xi + \alpha J \quad (3)$$

where  $\alpha$  is the combination coefficient.  $R_{w_h}$  contains all the relationships related to  $w_h$ . Our goal is to maximize the combined objective  $D$ .

Traditional neural networks often define the conditional probability  $p(y|x)$  in *softmax* function, which is impractical in this task due to the high cost of computing  $\nabla \log p(y|x)$  in the case of having hundreds of words in the vocabulary ( $10^5 - 10^7$  terms). In training process, we use negative sampling (NEG) [13] to solve this problem.

### 3.3 Join Relational Knowledge with Captioning Model

In general image captioning model, we often use CNN to extract image features and use RNN to combine the image feature with the corresponding caption. In this work, we adopt the Multimodal RNN mentioned in [6] as the captioning model, and we use a pretrained VGGNet [17] to extract spatial image features. Furthermore, by combining relational knowledge with the captioning model, our method attains the state-of-the-art performance.

We get the output word vectors  $E$  from the relational knowledge embedding model described in Sect. 3.2, and then use the  $E_t$  to represent the input vector  $x_t$  of the multimodal RNN, where  $E_t$  is the word encoding of the input word at timestep  $t$ . Besides the  $x_t$ , the multimodal RNN also takes the image pixels during training. It computes a sequence of outputs  $(y_1, \dots, y_t)$  by iterating the following recurrence relation:

$$b_v = W_{hi}[CNN(I)] \quad (4)$$

$$h_1 = f(W_{hx}x_1 + b_h + b_v) \quad (5)$$

$$h_t = f(W_{hx}x_t + W_{hh}h_{t-1} + b_h) \quad t > 1 \quad (6)$$

$$y_t = softmax(W_{oh}h_t + b_o) \quad (7)$$

where  $W_{hi}$ ,  $W_{hx}$ ,  $W_{hh}$ ,  $W_{oh}$ ,  $b_h$  and  $b_o$  are learnable parameters, and  $CNN(I)$  is the last layer of a CNN. We provide the image context vector  $b_v$  to the RNN only at the first iteration, which has been proven work better than at each time step in [6].

## 4 Experimental Results and Discussion

### 4.1 Datasets

To evaluate our proposed image captioning model, we experiment with MSCOCO [10] datasets. It contains 123,287 images, and we use the publicly available Karpathy splits [6] that have been used extensively in prior work to report our results. We get 113,287 images for training, 5,000 images respectively for validation and testing. Each image is annotated with 5 sentences.

We convert all sentences to lower case, discard non-alphanumeric characters and filter words whose frequency less than 5 in the training set, resulting in 9,488 words for training. We report our results using the standard automatic evaluation metrics, BLEU [15], METEOR [3], ROUGE-L [9] and CIDEr [18].

### 4.2 Evaluation

To verify the effectiveness of relational knowledge, we evaluate our full model (*ICRK*) against DeepVS model as well as other state-of-the-art models on image captioning.

In training, we encode the full-size input image with VGGNet [17] and set the size of hidden layer of RNN and the size of the input word embedding to 512, and we use Adam [7] algorithm to do model updating with an initial learning rate of  $4 \times e^{-4}$ .

Table 1 reports the performance of our ICRK which adds Skip-gram-RK to DeepVS relative to DeepVS baseline on the MSCOCO Karpathy test split. We also illustrate some qualitative captioning results of our model and the baseline in Fig. 2.

**Table 1.** Performance of our method on MSCOCO dataset

	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	ROUGE-L	CIDEr
LRCN [4]	62.8	44.2	30.4	21.0	-	-	-
DeepVS [6]	62.5	45.0	32.1	23.0	19.5	-	66.0
Our baseline: DeepVS	64.3	45.3	31.7	22.9	20.5	46.7	69.7
Our model: ICRK	<b>65.9</b>	<b>47.2</b>	<b>33.1</b>	<b>23.7</b>	<b>21.1</b>	<b>47.9</b>	<b>73.0</b>



DeepVS: a kitchen with a refrigerator and a stove  
Our ICRK: a refrigerator filled with lots of food and drinks  
Human: a refrigerator filled with lots of soft drinks



DeepVS: a man standing next to a fire hydrant  
Our ICRK: a parking meter sitting on the side of a road  
Human: series of parking meters and cars are located next to each other



DeepVS: two zebras are standing in a field of grass  
Our ICRK: two zebras standing next to each other in a zoo  
Human: zebras standing behind the fence in a zoo



DeepVS: a tennis player in action on the court  
Our ICRK: a tennis player is getting ready to serve the ball  
Human: a woman in a skirt gets ready to hit a tennis ball



DeepVS: a plate of food with a sandwich and salad  
Our ICRK: a white plate topped with meat and vegetables  
Human: a white plate with a variety of meat and vegetables



DeepVS: an elephant is standing in the middle of a field  
Our ICRK: a group of elephants standing next to each other  
Human: a group of elephants walking in muddy water.

**Fig. 2.** Qualitative captioning results of our method and DeepVS baseline. The descriptions generated by our model are more accurate than the descriptions generated by DeepVS, and our model combined with relational knowledge can recognize more reliable objects in image and make a better description.

## 5 Conclusion

In this paper, we present a novel image captioning model which combines relational knowledge with captioning model. Qualitative evaluation suggest that using relational knowledge as regularization function to learning word representations effectively improves the performance of image captioning model. Compared this method with two captioning baseline models and other works, our method achieves state-of-the-art performance.

**Acknowledgments.** This work was supported by National Key Research and Development Program of China (Grant No. 2016YFB1000902), National Program on Key Basic Research Project (973 Program, Grant No. 2013CB329600), and National Natural Science Foundation of China (Grant No. 61472040).

## References

1. Bollacker, K., Evans, C., Paritosh, P., Sturge, T., Taylor, J.: Freebase: a collaboratively created graph database for structuring human knowledge. In: Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, SIGMOD 2008, pp. 1247–1250 (2008)
2. Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. In: Advances in Neural Information Processing Systems, pp. 2787–2795 (2013)
3. Denkowski, M., Lavie, A.: Meteor universal: language specific translation evaluation for any target language. In: Proceedings of the Ninth Workshop on Statistical Machine Translation, pp. 376–380 (2014)
4. Donahue, J., et al.: Long-term recurrent convolutional networks for visual recognition and description. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2625–2634 (2015)
5. Fang, H., et al.: From captions to visual concepts and back. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2015
6. Karpathy, A., Fei-Fei, L.: Deep visual-semantic alignments for generating image descriptions. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2015
7. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. arXiv preprint [arXiv:1412.6980](https://arxiv.org/abs/1412.6980) (2014)
8. Lebet, R., Pinheiro, P.O., Collobert, R.: Simple image description generator via a linear phrase-based approach. In: ICLR (2015)
9. Lin, C.Y.: Rouge: a package for automatic evaluation of summaries. In: Text Summarization Branches Out (2004)
10. Lin, T.-Y., et al.: Microsoft COCO: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014, Part V. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). [https://doi.org/10.1007/978-3-319-10602-1\\_48](https://doi.org/10.1007/978-3-319-10602-1_48)
11. Lu, J., Xiong, C., Parikh, D., Socher, R.: Knowing when to look: adaptive attention via a visual sentinel for image captioning. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
12. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. In: ICLR (2013)

13. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J.: Distributed representations of words and phrases and their compositionality. *Adv. Neural Inf. Process. Syst.* **26**, 3111–3119 (2013)
14. Miller, G.A.: Wordnet: a lexical database for the english language. *Commun. ACM* **38**(11), 39–41 (2002)
15. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: BLEU: a method for automatic evaluation of machine translation. In: *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pp. 311–318. Association for Computational Linguistics (2002)
16. Rennie, S.J., Marcheret, E., Mroueh, Y., Ross, J., Goel, V.: Self-critical sequence training for image captioning. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2017)
17. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint [arXiv:1409.1556](https://arxiv.org/abs/1409.1556) (2014)
18. Vedantam, R., Lawrence Zitnick, C., Parikh, D.: CIDER: consensus-based image description evaluation. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4566–4575 (2015)
19. Xu, C., et al.: RC-NET: a general framework for incorporating knowledge into word representations. In: *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pp. 1219–1228. ACM (2014)
20. Xu, K., et al.: Show, attend and tell: Neural image caption generation with visual attention. *ICML* (2015)



# An Elite Group Guided Artificial Bee Colony Algorithm with a Modified Neighborhood Search

Jiaxin Lu, Xinyu Zhou<sup>(✉)</sup>, Yong Ma, and Mingwen Wang

School of Computer and Information Engineering,  
Jiangxi Normal University, Nanchang 330022, China  
xyzhou@whu.edu.cn

**Abstract.** Artificial bee colony (ABC) algorithm is a powerful global optimization method. For some complex optimization problems, however, ABC also suffers from a slow convergence speed due to its solution search equation which has strong exploration ability but poor exploitation ability. To solve the defect, in this paper, we proposed an elite group guided ABC algorithm with a modified neighborhood search operator, which aims to utilize the valuable information of elite individuals to guide search. First, some food sources with good fitness values are chosen as the elite individuals, and then they are used to construct an elite group. Second, in the onlooker bee phase, a novel solution search equation is designed based on the elite group, which introduces a parameter *MR* (modification rate) to control the greediness degree of the elite group guidance. Last, a modified neighborhood search operator is proposed based on the elite group, which is exploited to produce fine search in the vicinity of the elite individuals for a better tradeoff between exploration and exploitation abilities. In the experiments, 22 well-known test functions were used. The experimental results compared with other five ABC variants showed that our approach can achieve better or at least comparable performance on most of the test functions.

**Keywords:** Artificial bee colony algorithm · Solution search equation  
Elite group · Neighborhood search

## 1 Introduction

Artificial bee colony (ABC) algorithm [1] is a very effective and efficient swarm intelligence algorithm, developed by Karaboga in 2005 based on simulating the foraging behavior of a honeybee swarm. However, similar to these EAs, ABC also suffers from the defect that it has a slow convergence speed for some complex optimization problems. This is mainly caused by its solution search equation which has strong exploration ability but poor exploitation ability.

Over the past decade, many different ABC variants have been proposed to improve the performance of ABC. In order to improve the convergence speed, Akey and Karaboga [2] proposed a modified solution search equation in which a control parameter to control the frequency of parameter change. Zhu and Kwong [3] proposed

a global best (gbest)-guided ABC (GABC) by modifying the solution search equation, in which the valuable information of the gbest solution is incorporated into the solution search equation to enhance the exploitation. Be inspired by DE, Gao et al. [4] proposed a modified artificial bee colony algorithm (MABC). Furthermore, by introducing a global neighborhood search operator, Zhou et al. [5] proposed an improved MABC variant (MABC-NS). Cui et al. [6] proposed a novel depth-first search framework for ABC (DFSABC-elite).

In many improved ABC variants, the global best individual is severed as the source for providing valuable information. In such mechanisms, the performance of ABC may tend to be too greedy. To solve this defect, in this paper, we propose an elite group guided ABC algorithm with a modified neighborhood search (abbreviated as ENABC). In ENABC, we attempt to utilize a group of elite individuals for simultaneous guidance, which is helpful to increase the diversity for ABC while without losing the exploitation ability. First, some food sources with good fitness values are chosen as the elite individuals, and then they are used to construct an elite group. Second, in the onlooker bee phase, a novel solution search equation is designed based on the elite group, which introduces a parameter  $MR$  (modification rate) to control the greediness degree of the elite group guidance. Last, a modified neighborhood search operator is proposed based on the elite group, which is exploited to produce fine search in the vicinity of the elite individuals for a better tradeoff between exploration and exploitation abilities. In the experiments, 22 well-known test functions were used. The experimental results compared with other five ABC variants showed that our approach can achieve better or at least comparable performance on most of the test functions.

## 2 Original ABC

The original ABC includes four stages: initialization stage, employed bee stage, onlooker bee stage, and scout bee stage. After the initialization stage, ABC enters a circle of employed bee stage, onlooker bee stage, and scout bee stage until constraint condition is met. In initialization stage, the population is generated according to Eq. (1), which contains  $SN$  food sources,

$$x_{i,j} = x_j^{min} + rand(0, 1) * (x_j^{max} - x_j^{min}) \quad (1)$$

where  $i \in \{1, 2, \dots, SN\}; j \in \{1, 2, \dots, D\}$ .  $SN$  is the number of food source or employed bee or onlooker bee;  $D$  is the dimensionality of the search area;  $x_j^{min}$  and  $x_j^{max}$  represents the upper bound and the lower bound of the  $j$ th dimension.

After initialization stage, each employed bee will generates a new food source  $V_i$  according to Eq. (2),

$$V_{i,j} = x_{i,j} + \emptyset_{i,j} * (x_{i,j} - x_{k,j}) \quad (2)$$

where  $k \in \{1, 2, \dots, SN\}$  and has to be different from  $i; j \in \{1, 2, \dots, D\}$  is a randomly chosen dimension.  $\emptyset_{i,j}$  is a random number in the range  $[-1, 1]$ . If  $V_{i,j}$  exceed the



bound, they will be random generate in the bound. If the new food source  $V_i$  is better than its parent  $X_i$ , and then  $X_i$  is replaced with  $V_i$ , and its counter is reset to 0. Else the old one remain unchanged and its counter is increased by 1.

After all the employed bees have updated the position, they share the information of food source with onlooker bees. Each onlooker bee choose a food source depending on the probability value  $p_i$  associated with that food source, where

$$fit_i = \begin{cases} \frac{1}{1+f(x_i)} & \text{if } (f(x_i) \geq 0) \\ 1 + abs(f(x_i)) & \text{otherwise} \end{cases} \tag{3}$$

$$p_i = \frac{fit_i}{\sum_{j=1}^{SN} fit_j} \tag{4}$$

and  $f(x_i)$  is the objective function value of the  $i$ th food source and  $fit_i$  is the fitness value of the  $i$ th food source. When onlooker bee select a food source, they product a new food source  $V_i$  according to Eq. (2). The greedy selection method is employed to retain a better one from the old one and the modified one as well.

If a position cannot find a position better than current position at a limit number of times, this food source is considered to be exhausted. Then the food source has to be abandoned, and a new food source is generated according to Eq. 1.

### 3 Our Approach

#### 3.1 Motivations

Similar to other EAs, ABC also tends to show its unsatisfied performance when solving complex problems. Compare with the current individual, the elite individual is better than the current individual in most cases. Exploiting the information of the elite individual could degrade the exploration ability. In original ABC, while producing a new solution  $V_i$ , changing only one parameter of the parent solution  $x_i$  results in a slow convergence speed. In order to overcome the issue, Bahriye Akey and Devis Karaboga [2] introducing a control parameter to control the greediness degree. If a random number  $R_{ij}$  is less than MR, the parameter  $x_{ij}$  is modified as in the Eq. (5). It is noted that the control parameter  $MR$  is both used in employed bee and onlooker bee. Also this operator can accelerate the convergence speed, due to the solution search equation had no changed, the modified algorithm has strong exploration ability but poor exploitation ability too.

$$V_{i,j} = \begin{cases} x_{i,j} + \emptyset_{ij} * (x_{i,j} - x_{k,j}), & \text{if } R_{ij} < MR \\ x_{i,j}, & \text{otherwise} \end{cases} \tag{5}$$

### 3.2 An Elite Group Guided Multi-dimension Search Strategy

In order to enhance the exploitation ability and accelerate the convergence speed of ABC, we utilize a group of elite individuals for simultaneous guidance, and propose two novel solution equations:

$$V_{ij} = x_{r_1} + \emptyset_{i,j} * (x_{r_1} - x_{r_2}) \quad (6)$$

$$V_i = \begin{cases} x_e + \emptyset_{ij} * (x_e - x_i), & \text{if } R_{ij} < MR \\ x_i, & \text{otherwise} \end{cases} \quad (7)$$

where  $x_{r_1}$  and  $x_{r_2}$  are two individuals which randomly select from the population, and both different from  $x_i$ .  $\emptyset_{i,j}$  is the uniformly distributed random number in the range of  $[-1, 1]$ . And  $x_e$  is randomly pick up from the elite group. It is note that we select the top  $q * SN$  individuals in the fitness value to build the elite group in current population, and  $q$  set to 0.1 beforehand.  $R_{ij}$  is a random number in the range  $(0,1)$ .  $MR$  is a control parameter and we initialize it to 0.5.

The effectiveness of the Eq. (6) had been proved in Gao et al. [7] Because the vectors for generating candidate solution are all selected from the population randomly, the new search equation Eq. (6) has no bias to any search direction and with the exploration ability of ABC is improved significantly. The candidate solution  $V_i$  generated by Eq. (7) learns from randomly selected elite individual  $x_e$ , and combined with the  $MR$  operator, meanwhile. Compare with the search equation of original ABC, the new solution strategy draw lessons from the elite individual information and the parent information. And a lower value of  $MR$  may cause solutions to improve slowly while a higher one may cause too more diversity in a solution and hence in the population. In our proposed algorithm, Eq. (6) is used in the employed bee stage, and Eq. (7) is employed in the onlooker bee stage.

### 3.3 A Modified Neighborhood Search Operator

Although the search equation of NS shows good performance in MABC-NS [5], it also has some deficiency. The global best individual is treated as a reference for providing valuable information, in such mechanisms, it may tend to be too greedy. Two randomly individuals participate in search process, when the individuals had a bad position information, the solution search equation may showed unstable. To solve these problems, we proposed a modified neighborhood search based on elite group. The new search equation as Eq. (8),

$$TX_i = r_1 * X_i + r_2 * X_{e1} + r_3 * (X_{e2} - X_{e3}) \quad (8)$$

Be similar to original NS, where  $r_1, r_2$  and  $r_3$  are three non-negative numbers which are randomly chosen from  $(0, 1)$ , and they have a constraint condition that  $r_1 + r_2 + r_3 = 1$ .  $X_{e1}, X_{e2}$  and  $X_{e3}$  are three randomly selected food sources in elite group, and they are different from  $X_i$ .

The modified equation select three individuals from the elite group, which is exploited to produce fine search in the vicinity of the elite individuals for a better tradeoff between exploration and exploitation abilities. Compare with the global best join in search process, the operator used elite individual increase the diversity for algorithm while without losing the exploitation ability. Also the elite individual had better position than random individual in most cases,  $X_i$  the modified neighborhood search will show better stability than the old one. Noteworthy, we also employed a control parameter  $p$  to control the probability of use the neighborhood search operator. According to the experience of the MABC-NS [5],  $p$  set to 0.1.

### 3.4 Pseudo-code of ENABC

Compared with original ABC, ENABC make three modifications. First, we used two new search equations to replace the original search equations. Second, after the end of a generation, we used the modified neighborhood search operation to all individuals. Note also that we initialize all individual which the unchanged times has more than limit. The pseudo-code of ENABC is described in Algorithm 1, where  $FES$  is the number of used fitness function evaluations, and  $MaxFES$  as the stopping criterion, is the maximal number of fitness function evaluations.  $trial_i$  records the unchanged times of  $X_i$  fitness value.  $r_i$  is a random number between 0 and 1.

---

#### Algorithm 1 pseudo-code of ENABC

---

1. Randomly generate  $SN$  candidate solutions  $\{ X_i \mid i = 1, 2, \dots, SN \}$  as food source
  2. **While**  $FES \leq MaxFES$  **do**
  3.   **for**  $i = 1$  to  $SN$  **do** /\* employed bee stage \*/
  4.     Generate a new candidate solution  $V_i$  according to Eq. (6);
  5.   **end for**
  6.   Calculate the probability  $p_i$  according to Eq. (4); /\* onlooker bee stage \*/
  7.   **for**  $i = 1$  to  $SN$  **do**
  8.     Choose a food source  $X_j$  from the current population  $P$  by the roulette wheel selection mechanism;
  9.     Choose a food source  $X_e$  from the elite group;
  10.    Generate a new candidate solution  $V_i$  according to Eq. (7);
  11.   **end for**
  12.   **for**  $i = 1$  to  $SN$  **do** /\* scout bee stage \*/
  13.     **if**  $trial_i > limit$  **then**  $trial_i$
  14.       Randomly generate a new candidate solution  $X_i$  according to Eq. (1)
  15.     **end if**
  16.   **end for**
  17.   **for**  $i = 1$  to  $SN$  **do** /\* neighborhood search operator \*/
  18.     **if**  $r_i < p$  **then**
  19.       Generate a new candidate solution  $TX_i$  by Eq. (8);
  20.     **end if**
  21.   **end for**
  22. **end while**
-

## 4 Experimental Verifications

### 4.1 Benchmark Functions and Parameter Settings

A set of 22 benchmark functions is used to verify the performance of ENABC in the experiments, these problems are also widely used in other work [5]. Due to the limited space, we will just briefly describe these functions. Among these problems, the first 11 functions are unimodal problems, while the remaining ones are multimodal problems. The global optimum of all these problems is 0. Specifically, F05 is the Rosenbrock function which is multimodal when  $D > 3$ , F06 is a step function which has one minimum and is discontinuous, and F07 is a noisy quartic function. In the experiments, the corresponding maximum number of function evaluations (MaxFEs) is set to  $5000 \cdot D$ . Each benchmark function is run 30 times.

### 4.2 Comparison with Other ABC Variants

In this subsection, we present a comparative study between ENABC and five other ABC variants. These five compared ABC variants are listed as follows: ABC [1], MEABC [8], GBABC [9], MABC-NS [5] and AABC [10].

In order to ensure fairness, the food source numbers of all compared ABC algorithms are set to 75. The other control parameters of these algorithms are set according to their original papers. In ENABC, the control parameter  $MR$  is set to 0.5, the probability  $p$  of neighborhood search is set to 0.1,  $limit$  equals to 100. We presents a comparative study of ENABC with other ABC variants at both  $D = 30$  and 50. Due to the limited space, we only put  $30D$  experimental data in the article. The final results are given in Table 1, respectively.

**Table 1.** Experimental results of other four ABC variants and ENABC at  $D = 30$

Function	ABC	MABC_NS	MEABC	GBABC	AABC	ENABC
F01	2.80E-10-	1.63E-81-	6.46E-26-	2.05E-26-	2.72E-24-	3.95E-183
F02	9.97E-07-	6.25E-42-	3.08E-14-	3.48E-16-	3.33E-13-	5.55E-93
F03	7.32E+03-	3.72E-59-	9.34E+03-	2.84E+03-	9.95E+03-	1.16E-95
F04	3.13E+01-	1.03E-30-	9.56E+00-	6.70E-01-	2.18E+01-	3.44E-70
F05	7.68E-01+	2.78E+01-	1.55E+00+	2.34E+01≈	9.91E-01+	2.40E+01
F06	0.00E+00≈	0.00E+00≈	0.00E+00≈	0.00E+00≈	0.00E+00≈	0.00E+00
F07	1.78E-01-	1.41E-04+	3.79E-02-	2.32E-02-	9.31E-02-	5.55E-04
F08	1.04E-04-	3.47E-78-	5.74E-23-	1.64E-18-	1.17E-20-	7.71E-177
F09	1.04E-11-	2.29E-82-	6.40E-27-	3.90E-25-	2.76E-25-	2.97E-185
F10	3.20E-11-	8.74E-72-	2.07E-32-	3.76E-50-	4.77E-20-	2.57E-250
F11	7.47E-07-	0.00E+00+	1.24E-06-	1.52E-06-	1.55E-06-	4.22E-16
F12	3.82E-04≈	3.82E-04≈	3.82E-04≈	3.82E-04≈	3.82E-04≈	3.82E-04
F13	2.87E-09-	0.00E+00≈	0.00E+00≈	2.50E-03-	0.00E+00≈	0.00E+00
F14	3.51E-06-	4.44E-16+	4.08E-13-	1.03E-10-	1.32E-12-	1.75E-15

(continued)

**Table 1.** (continued)

Function	ABC	MABC_NS	MEABC	GBABC	AABC	ENABC
F15	5.24E-09-	0.00E + 00≈	0.00E+00≈	8.81E-07-	1.29E-12-	0.00E+00
F16	6.38E-12-	4.95E-31-	4.34E-28-	9.03E-26-	2.47E-26-	1.57E-32
F17	4.14E-10-	1.63E-29-	1.38E-26-	3.33E-24-	5.98E-25-	1.35E-32
F18	0.00E+00≈	0.00E+00≈	0.00E+00≈	0.00E+00≈	0.00E+00≈	0.00E+00
F19	7.91E-05-	6.85E-43-	6.28E-14-	1.43E-07-	2.86E-10≈	3.56E-93
F20	3.30E-08-	5.44E-28-	1.30E-25-	7.04E-23-	8.14E-17-	1.35E-31
F21	5.10E-08-	0.00E+00≈	0.00E+00≈	0.00E+00≈	0.00E+00≈	0.00E+00
F22	3.23E-04-	0.00E+00≈	0.00E00≈	2.15E-13-	3.20E-13-	0.00E+00
+/~!-	1/3/18	3/7/12	1/7/14	0/5/17	1/5/16	

For  $D = 30$ , from the results presented in Table 1, it is obvious that our approach achieves the best overall performance among the involved five algorithms. In the first 11 unimodal functions, ENABC have best performance than other ABCs variants on the majority of test functions, except Rosenbrock problem (F05). Note that on the Quartic with noise problem(F07) and Exponential problem(F11), MABC-NS had find the best value compare with other ABCs. Focus on the multimodel functions, as we can see that ENABC also has better performance. In the Schwefel 2.26 function (F12), NCRastrigin function (F18) and Bohachevsky\_2 (F21), all ABCs variants find a same value. And in the Rastrigin function and Griewank function, MABC-NS, MEABC and ENABC achieve the global optimum which are much better than other two algorithms. ENABC is better than other algorithms in the Generalized penalized 1 function, Generalized penalized 2 function, Alpine function and Levy function. The Wilcoxon’s rank sum test is conducted on the experimental results at a 5% significant level. It is noted that symbols “-”, “+”, “≈” denote that the performance of the corresponding algorithm is worse than, better than and similar to that of ENABC.

The Friedman test is also conducted to obtain the average rankings for both  $D = 30$  and 50, and Table 2 gives the final results. As seen, ENABC achieves the best average ranking. It means that ENABC is the best one among the six algorithms.

**Table 2.** Average rankings of all the six ABC variants at both  $D = 30$  and 50, and the best value is shown in **boldface**

Algorithms	Average rankings	
	$D = 30$	$D = 50$
ABC	5.11	5.43
MEABC	3.32	3.20
MABC-NS	2.32	2.18
GBABC	4.11	4.25
AABC	4.23	4.16
ENABC	<b>1.91</b>	<b>1.77</b>

## 5 Conclusions

In order to utilize the valuable information of elite individuals to guide search, in this paper we proposed an elite group guided artificial bee colony algorithm with a modified neighborhood search named ENABC, and utilize a group of elite individuals for simultaneous guidance, which is helpful to increase the diversity for ABC while without losing the exploitation ability. In ENABC, some food sources with good fitness values are chosen as the elite individuals, and then they are used to construct an elite group. Then, in the onlooker bee phase, a novel solution search equation is designed based on the elite group, which introduces a parameter  $MR$  (modification rate) to control the greediness degree of the elite group guidance. Last, a modified neighborhood search operator is proposed based on the elite group, which is exploited to produce fine search in the vicinity of the elite individuals for a better tradeoff between exploration and exploitation abilities. In the experiments, 22 well-known test functions were used. The experimental results compared with other five ABC variants showed that our approach can achieve better or at least comparable performance on most of the test functions.

**Acknowledgments.** This work is supported by the National Natural Science Foundation of China (Nos. 61603163, 61462045 and 61562042) and the Science and Technology Foundation of Jiangxi Province (No. 20151BAB217007).

## References

1. Karaboga, D., Basturk, B.: A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *J. Global Optim.* **39**(3), 459–471 (2007)
2. Akay, B., Karaboga, D.: A modified artificial bee colony algorithm for real-parameter optimization. *Inf. Sci.* **192**, 120–142 (2012)
3. Zhu, G., Kwong, S.: Gbest-guided artificial bee colony algorithm for numerical function optimization. *Appl. Math. Comput.* **7**(217), 3166–3173 (2010)
4. Gao, W., Liu, S.: A modified artificial bee colony algorithm. *Comput. Oper. Res.* **39**(3), 687–697 (2012)
5. Zhou, X., Wang, H., Wang, M., Wan, J.: Enhancing the modified artificial bee colony algorithm with neighborhood search. *Soft. Comput.* **21**(10), 2733–2743 (2017)
6. Cui, L., Li, G., Lin, Q.: A novel artificial bee colony algorithm with depth-first search framework and elite-guided search equation. *Inf. Sci.* **367–368**, 1012–1044 (2016)
7. Gao, W.F., Liu, S.Y., Huang, L.L.: A novel artificial bee colony algorithm based on modified search equation and orthogonal learning. *IEEE Trans. Cybern.* **3**(43), 1011–1024 (2013)
8. Wang, H., Wu, Z., Rahnamayan, S.: Multi-strategy ensemble artificial bee colony algorithm. *Inf. Sci.* **279**, 587–603 (2014)
9. Zhou, X., Wu, Z., Wang, H., Rahnamayan, S.: Gaussian bare-bones artificial bee colony algorithm. *Soft. Comput.* **20**, 907–924 (2016)
10. Yu, W., Zhan, Z., Zhang, J.: Artificial bee colony algorithm with an adaptive greedy position update strategy. *Soft. Comput.* **22**(2), 437–451 (2018)



# Exploiting Spatiotemporal Features to Infer Friendship in Location-Based Social Networks

Cheng He<sup>1</sup>, Chao Peng<sup>1(✉)</sup>, Na Li<sup>2</sup>, Xiang Chen<sup>3</sup>, and Lanying Guo<sup>1</sup>

<sup>1</sup> School of Computer Science and Software Engineering,  
East China Normal University, Shanghai, China  
chenghe28@qq.com, cpeng@sei.ecnu.edu.cn, 51164500097@stu.ecnu.edu.cn

<sup>2</sup> School of Data Science and Engineering,  
East China Normal University, Shanghai, China  
nali0606@foxmail.com

<sup>3</sup> School of Computer Science and Technology,  
Nantong University, Nantong, China  
xchencs@ntu.edu.cn

**Abstract.** The popularity of smart phone has brought the pervasiveness of location-based social networks. A large number of check-in data provides an opportunity for researchers to infer social ties between users. In this paper, we focus on three problems: (1) how to exploit fine-grained temporal features to characterize people's lifestyle. (2) how to use week-day and weekend check-ins data. (3) how to effectively measure the fine-grained location weight. To tackle these problems, we propose a unified framework STIF to infer friendship. Extensive experiments on two real-world location-based datasets show that our proposed STIF framework can significantly outperform the state-of-art methods.

**Keywords:** Social network · Social ties · Spatiotemporal features  
Location-based service · Inferring friendship

## 1 Introduction

In the past decade, social networks have become an essential part in our daily life. In particular, with the popularity of smart phones, people would like to post geo-tagged status and photos. Besides, in the location-based social networks (LBSNs) [3], users can share their location information (a.k.a check-in) when they find a new place or take part in social games. A large number of check-in data provides an opportunity for researchers to study various social behaviors. One interesting question is whether we can analyze users' social relationships based on their check-in data from LBSNs. In LBSNs, if two users are friend, they usually visit more same places compare to strangers, such as eating together at a restaurant. Based on this homophily principle, social ties, which are based on the difference of check-ins behavior between friends and non-friends, are studied in [8–10].

Inferring the social ties from those networks is of critical importance for social strength analysis, friend recommendation and targeted marketing.

In this paper, we propose a unified framework STIF to infer friendship. The main contributions of this paper are summarized as follows:

- To the best of our knowledge, we are the first to exploit the fine-grained temporal features to characterize peoples' lifestyles, mine the different trajectories on weekday and weekend check-ins, and measure the different weights of a certain location at different time periods.
- We propose a unified framework STIF to quantify the friendship, which combines the above spatiotemporal features as the input of machine learning algorithms after dealing with imbalanced data by sampling.
- We conduct extensive experiments to evaluate the performance of our proposed framework on two real-world datasets and the experiment results show the superiority of our framework when comparing to state-of-the-art methods.

The paper is organized as follows. We summarize related work in Sect. 2. In Sect. 3, we present the definitions and our framework STIF. Section 4 describes our proposed spatiotemporal features. Section 5 reports experiment result. We conclude this paper and discuss future work in Sect. 6.

## 2 Related Work

Inferring friendship from the location-based check-in data has been a hot research topic in the past few years. Scellato et al. [9] defined extensive new prediction features based on the properties of the places visited by users, and then they modelled inferring friendship as a binary classification problem. Zhang et al. [11] discovered that friends' frequent movement areas are closer than strangers, they utilized several distance metrics to quantify the distance of two users' frequent movement areas for inferring friendship in LBSNs.

It is intuitive that more frequently two peoples meet, the stronger their mobility relationship is. However, Wang et al. [10] argued that not all the *co-occurrence* (a.k.a meet at a same location) are equally important in measuring the relationship, so they considered personal factors, global factors and temporal factors to differentiate the *co-occurrence*. Similar to [10], Njoo et al. [7] explored the features including diversity feature, temporal stability feature and duration feature from *co-occurrence* to infer friendship. Zhao et al. [12] believed that a user should have a specific intention at one trip and each specific intention corresponds to a hidden Markov model, which can mine user's implicit behavior patterns.

Different from previous studies, we consider the situation that a certain location has different weights at different time periods. Moreover, we also consider a user's check-in habit, including the location where a user has most check-ins at different time periods and distance of co-occurrence of user pairs, etc.

## 3 The Proposed Framework

In this section, we introduce the preliminaries of STIF framework and the detail of STIF framework for inferring friendship.



### 3.1 Preliminaries

**Definitions:** A check-in tuple  $\langle u, l, t \rangle$  means a user  $u$  checks in a location  $l$  at the time  $t$ . Given a user  $u$ , his all check-ins is denoted by a sequence  $C_u = \{\langle u, l_1, t_1 \rangle, \dots, \langle u, l_n, t_n \rangle\}$ , where the sequence is ordered by time ascendingly.

**Friendship Inferring:** Given a check-in dataset  $C$  and  $u_1, u_2 \in U$  with the form of  $\langle u, l, t \rangle$ , the task is to predict whether  $u_1$  and  $u_2$  are friends or not.

**Co-occurrence:** We say that  $u_1$  and  $u_2$  have a *co-occurrence* at location  $l$  if their check-in distance is less than distance threshold  $\delta$  and the time difference is less than time threshold  $\tau$ , where the parameters  $\delta$  and  $\tau$  are application-dependent and can be set experimentally. In our paper, we set  $\delta = 500$  m and  $\tau = 2$  h.

Let  $m_{u_1, u_2}^{l, t} = \langle u_1, u_2, l, t \rangle$  denotes that  $u_1$  and  $u_2$  have a *co-occurrence* in location  $l$  at time  $t$ , where the co-occurrence time  $t$  is calculated by averaging the time of  $u_1$  and  $u_2$ , co-occurrence location  $l$  is the average value of two user's locations.  $M_{u_1, u_2} = \{m_{u_1, u_2}^{l_1, t_1}, \dots, m_{u_1, u_2}^{l_n, t_n}\}$  is the set of all *co-occurrence* of  $u_1$  and  $u_2$ . Therefore,  $|M_{u_1, u_2}|$  is the total number of *co-occurrence* of  $u_1$  and  $u_2$ .

### 3.2 Spatiotemporal Features to Infer Friendship(STIF)

As shown in Fig. 1, in our algorithm we first select the top-10000 users' check-in records (ranked by the number of check-ins) from the original dataset. The original dataset have lots of inactive users, according to our statistics in *Gowalla* dataset, the number of top-10000 users' check-ins has already accounted for 54% of the total check-ins. It means that the top-10000 users are the active users who have a large amount of check-ins, which is meaningful for inferring friendships. Secondly, we extract the spatiotemporal features described in Sect. 4 and label the samples according to the friendship (*ground truth*) of original dataset to generate the train set. Then, we take synthetic minority oversampling technique (SMOTE) [1] to increase the number of minority (a.k.a friendships) samples.

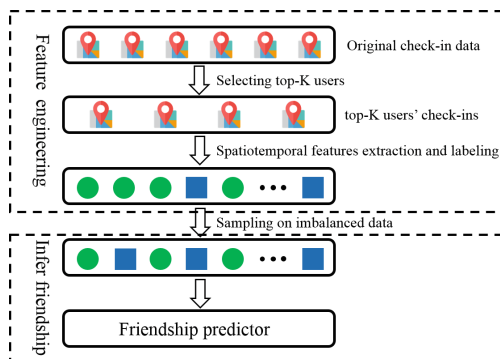


Fig. 1. The overview of STIF framework

Finally, we utilize the supervised machine learning algorithms to train the friendship predictor. We mainly consider five classical algorithms, i.e., Naive Bayes (NB), Neural Network (NN), K-Nearest Neighbor (KNN), Decision Tree (C4.5) and Random Forest (RF)<sup>1</sup>. All our evaluation tests have been performed with the WEKA framework, using default parameters.

## 4 Spatiotemporal Features

The key of inferring friendship is how to extract effective user representation from check-in data. We consider spatiotemporal features from four aspects, including: exploiting fine-grained temporal features, modeling weekday and weekend check-ins, measuring fine-grained location weight and co-occurrence features.

### 4.1 Exploiting Fine-Grained Temporal Features

According to the “homophily principle” [6], friends are more likely to have similar habits or lifestyles than strangers. For example, if two users like to check-in from 8 a.m to 9 a.m every day, it indicates they have similar temporal lifestyle. Let  $n_{u_1}^h = \{n_{u_1}^{h_0}, \dots, n_{u_1}^{h_{23}}\}$  and  $n_{u_1}^d = \{n_{u_1}^{d_1}, \dots, n_{u_1}^{d_7}\}$  denote the set for the number of check-ins per hour and per day, respectively. Note that we divide a day by 24 with each part being a one hour period and divide a week into 7 days. Then, we can get two features consisting of *hour\_sim* and *day\_sim* by the Cosine similarity:

$$hour\_sim(u_1, u_2) = \frac{n_{u_1}^h \cdot n_{u_2}^h}{\|n_{u_1}^h\| \cdot \|n_{u_2}^h\|}, \quad day\_sim(u_1, u_2) = \frac{n_{u_1}^d \cdot n_{u_2}^d}{\|n_{u_1}^d\| \cdot \|n_{u_2}^d\|} \quad (1)$$

In addition, we design another two temporal features of  $u_1$  and  $u_2$ : *hour\_diff*(*hd*) and *day\_diff*(*dd*), which can be formally defined as follows:

$$hd = \left| \arg \max_i n_{u_1}^{h_i} - \arg \max_j n_{u_2}^{h_j} \right|, \quad dd = \left| \arg \max_i n_{u_1}^{d_i} - \arg \max_j n_{u_2}^{d_j} \right| \quad (2)$$

The *hd* represents the difference in lifestyle between two users, for example,  $u_1$  may like to get up at 6 a.m while  $u_2$  may get up at 11 a.m, therefore,  $u_1$  and  $u_2$  have a big difference in terms of lifestyle. The function of *dd* is similar to *hd*.

### 4.2 Modeling Weekday and Weekend Check-ins

The geographic distance between two users is highly related to social ties. We follow the idea in [9] and employ the “home-location” where user has most number of check-ins to compute the geographic distance between two users’ home locations. Beyond that, we divide the user’s check-ins into weekday check-ins and weekend check-ins based on the consideration that people have great difference in lifestyles on weekday and weekend. Then, we compute the distance

<sup>1</sup> Naive Bayes corresponds to Naive Bayes, Neural Network corresponds to Multilayer perceptron (MLP), KNN corresponds to IBK, Decision Tree (C4.5) corresponds to J48, Random Forest corresponds to Random Forest in WEKA, respectively.

of “weekday-location” (a.k.a the location that user has most check-ins on weekday) and distance of “weekend-location” between two users respectively. What’s more, we extract user’s trajectory from check-ins data to quantify the similarity of their check-ins trajectory by cosine similarity.

### 4.3 Measuring the Fine-Grained Location Weight

To measure the importance of a certain location, *location entropy* was widely used [2, 7–11], which was introduced in [4] and can be defined as follows:

$$location\_entropy(l) = - \sum_{u \in \Phi_l} (C_u^l / C^l) \log(C_u^l / C^l) \quad (3)$$

where  $\Phi_l$  represents all users that have checked in location  $l$ ,  $C_u^l$  is the check-in frequency of user  $u$  at location  $l$  and  $C^l$  represents the total number of check-ins, which all users have had at location  $l$ .

However, a same location may have different popularities at different time periods. For example, if  $u_1$  and  $u_2$  have a co-occurrence at the metro station during a morning rush hour. In contrast if  $u_1$  and  $u_3$  appear at the metro station in midnight when there are few people, we can infer that  $u_1$  and  $u_3$  are more likely to be friends. Based on the above considerations, we propose a fine-grained metric to quantify location popularity of different time periods:

$$location\_time\_popularity(l) = \beta e^{-C_t^l} \quad (4)$$

where  $C_t^l$  is the total number of check-ins in the period  $[t - 1, t + 1]$ .  $\beta$  is a popularity parameter for quantifying the popularity of a location, since different locations have different popularities. Therefore, parameter  $\beta$  can be defined as the reciprocal of *location\_entropy*( $l$ ).

In addition, when two users meet at the same place, how long they stay in this place is an important factor to indicate whether they are friends. We further take the time they stay together at a certain place into consideration. However, in practice, it is difficult to directly obtain the accurate stay time from the LBSNs data. We use the time interval between two consecutive co-occurrences to approximately estimate it.

Based on above analysis, we can measure the weight of a co-occurrence:

$$w^i(u_1, u_2) = \frac{location\_time\_popularity(l)}{location\_entropy(l) + \varepsilon} \times stay\_time_l \quad (5)$$

where  $\varepsilon$  is a smoothing term that avoids division by zero, we fix  $\varepsilon = e^{-10}$ . For a series of co-occurrences between two users, we can compute their co-occurrence score by summing  $w^i(u_1, u_2)$ .

### 4.4 Co-occurrence Features

It is obvious that two users are likely to be friends if their mean distance interval (i.e., *average distance interval*) of co-occurrence is small because it indicates that they often have co-occurrence in certain areas. In order to grasp the fine-grained characteristics of distance interval, we take the *maximum distance interval* and *minimum distance interval* into consideration.

## 5 Experiments

In this section, we conduct our experiments and report the experimental results.

### 5.1 Dataset

We conduct our experiment on two public real-world datasets, namely *Gowalla* and *Brightkite* [5]. Gowalla and Brightkite are two location-based social networking websites where users share their locations by check-in. The form of check-in record is:  $\langle user\_id, time, latitude, longitude, location\_id \rangle$ , the detailed statistics of the two datasets are given in Table 1.

**Table 1.** Statistics of Datasets

Dataset	# Users	# Check-ins	# Friend pairs (edges)
Gowalla	107,092	6,442,890	950,327
Brightkite	58,228	4,491,143	214,078

### 5.2 Performance Measures and Model Evaluation Method

To fully verify the effectiveness of our framework, we not only adopt the commonly used precision, recall and F1 as metrics, but also use AUC (area under the ROC curve) to evaluate the performance of our method. precision, recall and F1 can be formally defined as follows:

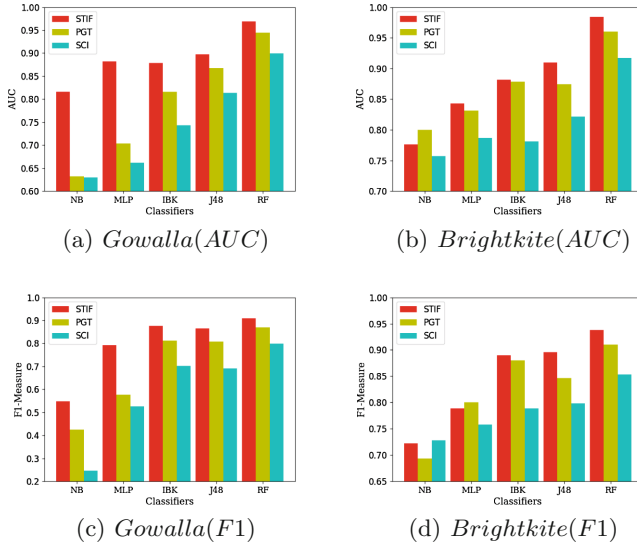
$$P = \frac{|TP|}{|TP| + |FP|}, R = \frac{|TP|}{|TP| + |FN|}, F1 = 2 \times \frac{P \times R}{P + R} \quad (6)$$

where  $TP, FP, TN$  and  $FN$  denote true positive, false positive, true negative and false negative, respectively. To confidently evaluate the performance of our friendship predictor, A ten-fold cross-validation was applied in our paper.

### 5.3 Baseline Methods

We compare the proposed model STIF with two state-of-the-art methods, including:

- PGT [10]: To measure the relationship between two given users, PGT extracts three features including *person factor*, *global factor* and *temporal factor* as the input of classification algorithms to infer friendship.
- SCI [7]: For each user pair, SCI mainly considers the *diversity* and *temporal feature* of co-occurrence. The *diversity* measure how diverse an co-occurrence is and the *temporal feature* quantifies the stability of an co-occurrences.



**Fig. 2.** Performance Comparisons for Inferring Friendship

#### 5.4 Experiment Result Analysis

We use NB, MLP, IBK, J48 and RF to train the friendship predictor on the two real-world datasets *Gowalla* and *Brightkite*, respectively. Table 2 shows the experiment results of baseline methods and our method based on precision, recall, F1 and AUC.

As shown in Fig. 2, we can find that our method STIF achieves the best performance compared to PGT and SCI by F1 and AUC, respectively. Specifically, in terms of F1, which is shown in Table 2, our method obtains the best performance when using random forests classifiers in both datasets, with the value of 0.908 in Gowalla dataset and 0.938 in Brightkite dataset, respectively. In Gowalla dataset, our method outperforms baseline methods in all five classifiers. Our method improves 30.2% over the SCI method in NB classifier, and 22% over the PGT method in MLP classifier. In Brightkite dataset, although our method perform slightly worse than NB by 0.6% and MLP by 1.1%, it is better than baseline methods in other classifiers, specifically, the performance of our method is 5.2% higher than PGT on J48 classifier.

From the perspective of AUC value, from Table 2 we can find that our method achieves a remarkable improvement compared to the two baseline methods. Furthermore, the AUC value of our method is improved by 17.8% and 22.0% compared to PGT and SCI on MLP in Gowalla dataset, respectively. In Brightkite dataset, our method achieves the best performance on all classifiers except for NB classifier.

**Table 2.** Precision, recall, F1 and AUC for different classifiers on the datasets

Classifier	Method	Gowalla				Brightkite			
		P	R	F1	AUC	P	R	F1	AUC
NB	STIF	<b>0.948</b>	<b>0.385</b>	<b>0.548</b>	<b>0.816</b>	0.657	0.801	0.722	0.776
	PGT	0.651	0.314	0.424	0.632	<b>0.785</b>	0.619	0.693	<b>0.800</b>
	SCI	0.664	0.151	0.246	0.630	0.631	<b>0.861</b>	<b>0.728</b>	0.757
MLP	STIF	<b>0.795</b>	<b>0.790</b>	<b>0.792</b>	<b>0.882</b>	0.737	0.850	0.789	<b>0.843</b>
	PGT	0.658	0.512	0.576	0.704	<b>0.751</b>	<b>0.856</b>	<b>0.800</b>	0.831
	SCI	0.632	0.448	0.525	0.662	0.741	0.776	0.758	0.787
IBK	STIF	<b>0.831</b>	<b>0.929</b>	<b>0.877</b>	<b>0.879</b>	<b>0.845</b>	<b>0.939</b>	<b>0.890</b>	<b>0.882</b>
	PGT	0.783	0.844	0.813	0.816	0.872	0.888	0.880	0.878
	SCI	0.738	0.671	0.703	0.743	0.797	0.782	0.789	0.781
J48	STIF	<b>0.860</b>	<b>0.872</b>	<b>0.866</b>	<b>0.897</b>	<b>0.879</b>	<b>0.913</b>	<b>0.896</b>	<b>0.910</b>
	PGT	0.774	0.843	0.807	0.867	0.825	0.870	0.847	0.874
	SCI	0.832	0.591	0.691	0.814	0.769	0.829	0.798	0.821
RF	STIF	<b>0.911</b>	<b>0.905</b>	<b>0.908</b>	<b>0.969</b>	<b>0.931</b>	<b>0.945</b>	<b>0.938</b>	<b>0.984</b>
	PGT	0.857	0.884	0.870	0.945	0.884	0.936	0.910	0.960
	SCI	0.875	0.735	0.799	0.899	0.849	0.857	0.853	0.917

## 6 Conclusion

In this paper, we have analyzed the problem of inferring friendship from users' check-in data in Location-based social network. To address this problem, we proposed the STIF method to mine users' spatiotemporal characteristics from their check-ins. We conduct extensive experiments on two real-world datasets and the result demonstrates the effectiveness of our STIF method and its superiority over state-of-the-art baseline methods. In the future, we plan to apply topic model to mine more latent similarities between friends based on co-occurrences.

## References

1. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: SMOTE: Synthetic minority over-sampling technique. *JAIR* **16**(1), 321–357 (2002)
2. Cheng, R., Pang, J., Zhang, Y.: Inferring friendship from check-in data of location-based social networks. In: *ASONAM*, pp. 1284–1291 (2015)
3. Cho, E., Myers, S.A., Leskovec, J.: Friendship and mobility: User movement in location-based social networks. In: *SIGKDD*, pp. 1082–1090 (2011)
4. Cranshaw, J., Toch, E., Hong, J., Kittur, A., Sadeh, N.: Bridging the gap between physical location and online social networks. In: *UbiComp*, pp. 119–128 (2010)
5. Leskovec, J., Krevl, A.: SNAP Datasets: Stanford large network dataset collection, Jun 2014. <http://snap.stanford.edu/data>
6. McPherson, M., Smith-Lovin, L., Cook, J.M.: Birds of a feather: Homophily in social networks. *Ann. Rev. Sociol.* **27**(1), 415–444 (2001)

7. Njoo, G.S., Kao, M.C., Hsu, K.W., Peng, W.C.: Exploring check-in data to infer social ties in location based social networks. In: PAKDD, pp. 460–471 (2017)
8. Pham, H., Shahabi, C., Liu, Y.: EBM: An entropy-based model to infer social strength from spatiotemporal data. In: SIGMOD, pp. 265–276 (2013)
9. Scellato, S., Noulas, A., Mascolo, C.: Exploiting place features in link prediction on location-based social networks. In: SIGKDD, pp. 1046–1054 (2011)
10. Wang, H., Li, Z., Lee, W.C.: PGT: Measuring mobility relationship using personal, global and temporal factors. In: ICDM, pp. 570–579 (2015)
11. Zhang, Y., Pang, J.: Distance and friendship: A distance-based model for link prediction in social networks. In: APWeb, pp. 55–66 (2015)
12. Zhao, W.X., Zhou, N., Zhang, W., Wen, J.R., Chang, E.Y., Chang, E.Y.: A probabilistic lifestyle-based trajectory model for social strength inference from human trajectory data. *TOIS* **35**(1), 8 (2016)



# A Subsequent Speaker Selection Method for Online Discussions Based on the Multi-armed Bandit Algorithm

Mio Kurii<sup>(✉)</sup> and Katsuhide Fujita

Tokyo University of Agriculture and Technology, Tokyo, Japan  
{kurii,fujita}@katfuji.lab.tuat.ac.jp

**Abstract.** This paper proposes a method to select subsequent speakers in an online discussion, which is one of the important functions of facilitators, using the multi-armed bandit algorithm. Bandit algorithms can be applied to speaker determination by considering each participant as an arm of a slot machine and a facilitator as a player. We define a “discussion score” to evaluate each post, and it is then considered to be equivalent to the reward of the slot machine method. The discussion score of each post is defined based on the following three metrics: (1) Whether the post helps to settle a discussion or not. (2) How interested are the other participants in the post (3) The intention of the post. To consider conflict between participants, our method classifies the participants into groups and determines the next speaker based on clustering results. We demonstrate that our method can select participants who posted good ideas and opinions and promote participants to engage other participants by using questionnaires.

**Keywords:** Multi-armed bandit problem  
Decision support system · Automated facilitator

## 1 Introduction

Online discussions are becoming increasingly prevalent owing to the popularity of smartphones and social networking services.

Online discussions have many advantages. For example, it is possible to conduct meetings regardless of the users’ locations and to record the history of conversations. However, this form of remote meeting method also has some disadvantages. Users have difficulties reading and conveying indirect communications, including gestures, nervous habits, room tension, and eye contact. Therefore, problems such as conversation delays and miscommunications can easily occur. One of suggested solutions is to develop an automated facilitator which manages a discussion. This paper focuses on the action in which subsequent speakers can be determined using a facilitator. The participant determination system is important for deciding who the facilitator will select as the next speaker.



The remainder of this paper is organized as follows. First, we discuss related research. Next, we propose the subsequent speaker determination method based on the multi-armed bandit algorithm. Then, we demonstrate our experimental results. Finally, we present our conclusions.

## 2 Subsequent Speaker Determination Method Based on the Multi-armed Bandit Algorithm

### 2.1 Applying the Bandit Algorithm to Speaker Determination

In this paper, we assume that speaker determination can be represented as a bandit problem and solve it with using UCB policy that is one of the most famous bandit algorithms, based on Kuleshov and Precup’s study [2]. Bandit problems are famous as slot machine problems, but also can be used for many types of optimization problems that need to analyze trade-offs between exploration and exploitation. Examples of these include game tree searches [6] and clinical trials [7]. The facilitator’s behavior when they select a participant and urge the participant to speak can be considered to be equivalent to the behavior of a player who selects one slot and plays it in the multi-armed bandit problem. Accordingly, the participants of a discussion can be regarded as slots. The reward of a slot action is either a hit or a miss (0 or 1). In speaker determination, we defined a score named the discussion score that reflects the influence of the post posted by a selected participant. We consider this score to be equivalent to a reward in the multi-armed bandit problem.

### 2.2 Discussion Score

A discussion score  $f_p$  of post  $p$  is defined based on three points as follows:

#### **Whether the post helps to solve the issues in a discussion or not ( $f_1$ ):**

We assume that the discussion system has “Agree” and “Disagree” buttons so that each participant can evaluate each post intuitively an independently, similar to a “Like” function in social media. This function enables people to view the ratio between “Agrees” and “Disagrees.” Posts that gain much support from participants can be considered to help in solving the issues in a discussion.  $f_1$  of post  $p$  based on this concept is defined as follows:

$$f_1(p) = \frac{N_{agree}(p) - N_{disagree}(p)}{K} \quad (1)$$

$N_{agree}(p)$  means the number of “Agree” that post  $p$  gets, so as  $N_{disagree}(p)$ .  $K$  is the number of participants.

**How interested are the other participants in the post ( $f_2$ ):** A post that attracts interest from participants can make a discussion active and influence it positively. It can be measured by the number of replies that other

participants sent to the post. Therefore,  $f_2$  of post  $p$  based on this concept is defined as follows:

$$f_2(p) = \frac{N_{reply}}{K} \quad (2)$$

$N_{reply}(p)$  means the number of replies to the post  $p$ .

**The intention of the post ( $f_3$ ):** The type of post is a very important factor in post categorization. Some online discussion systems have a category tagging function to reflect the statement intention. In this study, we defined seven types of categories; #proposal, #explanation, #supporting, #confronting, #question, #answer, and #etc, based on Kotani et al [1].

Posts categorized as “#proposals” are considered to have a good influence on a discussion. Additionally, posts categorized as “#explanations” are considered to increase the productivity of a discussion. Therefore, we define  $f_3(p) = 1$  if the category tag of post  $p$  is a “#proposal” or an “#explanation”, otherwise  $f_3(p) = 0$ .

The above three scores are weighted based on each of their importance.  $f_1$  can be considered to have the highest importance because it directly reflects a participant’s preference.  $f_2$  has the second highest importance because it depends on how active the discussion is.  $f_3$  is the lowest. Using this rank, the discussion score  $f(p)$  of post  $p$  is defined as follows.

$$f(p) = \frac{3}{6}f_1(p) + \frac{2}{6}f_2(p) + \frac{1}{6}f_3(p) \quad (3)$$

### 2.3 Clustering Using Bipartite Graph

Omoto [5] revealed the importance of relationships between participants, particularly conflict/cooperative relationships to make an agreement. When the bandit algorithm is applied to speaker determination, the relationships between participants are not considered. To solve this problem, we propose the use of a “two-step bandit application.” A “two-step” approach means that the participants are divided into groups, and they select one group first and then select one participant from the group. Here, we apply the bandit algorithm (UCB policy) for both group determination and speaker determination.

To apply the UCB policy to group determination, we consider each group to be a slot. To calculate the UCB score of group  $G$ , the following two values are necessary.

1. The average discussion scores from all the posts by the participants belonging to group  $G$
2. The total number of posts from participants belonging to group  $G$

To divide the participants into appropriate groups, participants with similar opinions should be put in the same group, while participants with contrasting opinions should be separated from each other. We applied the clustering method

using the bipartite graph that was used by Nakahara et al. [3] for tweet clustering. Then, the participants are clustered based on the “Agree” and “Disagree” overlap degree of each post between participants. This paper assumes the union of posts in which user  $U_i$  puts “Agree” is  $P_a(i)$  and user  $U_i$  puts “Disagree” is  $P_d(i)$ . The overlap degree of the agreement between user  $A$  and user  $B$  is calculated by the ratio of  $P_a(A) \cap P_a(B)$  to  $P_a(A) \cup P_a(B)$  (Jaccard similarity coefficient). This paper defines a similarity value  $O_{ij}$  between user  $U_i$  and user  $U_j$  as follows.

$$O_{ij} = \frac{|P_a(A) \cap P_a(B)| + |P_d(A) \cap P_d(B)|}{|P_a(A) \cup P_a(B)| + |P_d(A) \cup P_d(B)|} \quad (4)$$

The next step is the construction of a network graph based on the similarity values between participants. Each node represents one participant. If the similarity of two participants is larger than the threshold  $th$ , their nodes are linked. After this procedure, *best\_partition* is applied to the network graph. *best\_partition* is a clustering function included in *community* (a library of Python). We set the value of  $th$  to 0.45 based on a preliminary simulation using existing discussion data.

### 3 Experiments

The purpose of the experiment was to evaluate the validity and the usefulness of our method by applying it to real online discussions.

#### 3.1 Experimental Settings

We conducted three discussions with 21 subjects and three groups with seven people in each group. One group consisted of two discussions based on the following topics.

1. A foreign professor visits Japan next week for the first time in his life. Pick three types of sushi for this meal that you think he will like.
2. The 2020 Tokyo Olympics are underway. Pick three representative characters for Japan among the existing characters.

Each discussion lasted 75 mins, which were separated into the former (25 mins) and the latter (50 mins) parts. The former part was an open discussion. The latter part took place under the following rule: Each of the participants picked by a facilitator can post once (1 post only).” The facilitator conducted the experiment, and determined the next speaker among the participants according to the speaker determination system. After the selected participant posted or two minutes passed after the participant was selected, the facilitator selected the next speaker. This process was repeated until the time span elapsed. Before starting the experiments, the facilitator informed the participants about the goals of the discussion; they were “presenting as many ideas as possible” and “reaching a conclusion that every

participant can agree upon.” We used two speaker selection determination methods; one of them determined the next speaker based on the proposed method, and the other determined the next speaker randomly. Two different speaker determination approaches were used in separate experiments to compare them. Table 1 shows the topic and method was used in each discussion.

**Table 1.** The list of discussion’s conditions

	Group 1	Group 2	Group 3
Discussion 1	Topic 1	Topic 1	Topic 2
	Baseline Method	Proposed Method	Baseline Method
Discussion 2	Topic 2	Topic 2	Topic 1
	Proposed Method	Baseline Method	Proposed Method

We added two rules so that the discussion appeared to be natural.

- Both the proposed method and the baseline did not select the same participant two times in a row.
- In the UCB policy, the slot with the highest UCB score should be selected. In this experiment, we turned it into a stochastic selection system. The next speaker was selected randomly based on the expectation of each participant, which was in direct proportion to their UCB score. For example, if the ratio of the UCB score between the participants  $u_1$  and  $u_2$  is 5 : 1, the probability that the system picks  $u_1$  is five times as high as that of  $u_2$ .

A questionnaire about the experiment was conducted to evaluate our proposed method. The list of questions and their options are represented as follows. (1) There were many good ideas and opinions in the latter part of the discussion. (2) There were only limited opinions and ideas in the latter part of the discussion. (3) The other participants’ ideas and opinions contributed to your final opinion on the topic during the discussion. (4) Were the speakers appropriately determined by the facilitator? (5) Evaluate the other participants’ discussion abilities.

### 3.2 Results

Table 2 shows the totals from each discussion (The former and the latter). In all of them, the former parts had larger numbers of posts than the latter parts.

#### **There were many good ideas and opinions in the latter part of the discussion**

The options of this questions are (a) Strongly disagree, (b) Disagree, (c) Neither agree nor disagree, (d) Agree, and (e) Strongly agree. Table 3 shows the results of 21 subjects. The percentage of those from our proposed method who

**Table 2.** Total number of posts in each discussion

	Group 1	Group 2	Group 3
Discussion 1	67 (42:25)	72 (44:28)	119 (85:34)
Discussion 2	60 (36:24)	76 (47:29)	127 (97:30)

**Table 3.** There were many good ideas and opinions in the latter part of the discussion

	(a)	(b)	(c)	(d)	(e)
Baseline Method	0 (0.0%)	1 (4.8%)	1 (4.8%)	4 (19.0%)	1 (4.8%)
Proposed Method	0 (0.0%)	2 (9.5%)	2 (9.5%)	2 (9.5%)	1 (4.8%)

answered “Agree” (includes (d) and (e)) was 72%, even though the baseline was 62%. Each group’s result also showed consistent results (the proposed method’s percentage was larger than that of the baseline method). Thus, the proposed method can successfully select participants who posted good ideas and opinions. The UCB policy tended to select a slot that had not been picked for a sufficient number of times before. Participants who had not posted as much in a former part tended to be picked more than those who had posted a lot. In the latter part, those participants often posted interesting ideas and opinions from a new perspective.

### **There were only limited opinions and ideas in the latter part of the discussion**

The options of this questions are (a) Strongly disagree, (b) Disagree, (c) Neither agree nor disagree, (d) Agree, and (e) Strongly agree. This question is asked to confirm the effect of the clustering and the double bandit method. Table 4 shows the results of the question. The percentage of who answered “Disagree” (including (a) and (b)) in the proposed method was 10% larger than that of the baseline method (34%). The results of each group were also the same (the percentage of the proposed method was larger than that of the baseline). Therefore, the proposed method prevented the discussions from having only similar ideas and opinions and promoted diverse ideas and opinions.

### **The other participants’ opinions contributed to your final opinion on the topic**

The options of this questions are (a) Strongly disagree, (b) Disagree, (c) Neither agree nor disagree, (d) Agree, and (e) Strongly agree. This question was asked to confirm whether the proposed method can contribute to the productivity of the discussions. Table 5 shows the results. Exchanging ideas and opinions actively is one of the most important aspect of discussions. This question can be considered to be an efficient approach to measure it. Using the proposed method, the percentage of those who answered “Agree” (includes (d) and (e)) increased from 67% to 86%, compared to the baseline method. Owing to the above results, the proposed method inspired the participants

**Table 4.** There were only limited opinions and ideas in the latter part of the discussion

	(a)	(b)	(c)	(d)	(e)
Baseline Method	0 (0.0%)	2 (9.5%)	4 (19.0%)	13 (61.9%)	2 (9.5%)
Proposed Method	1 (4.8%)	6 (28.6%)	1 (4.8%)	11 (52.4%)	2 (9.5%)

**Table 5.** The other participants’ ideas and opinions contributed to your final opinion on the topic through the discussion

	(a)	(b)	(c)	(d)	(e)
Baseline Method	1 (4.8%)	2 (9.5%)	4 (19.0%)	6 (28.6%)	8 (38.1%)
Proposed Method	0 (0.0%)	3 (14.3%)	0 (0.0%)	9 (42.9%)	9 (42.9%)

to engage each other. These results are considered to be strongly related to the results of the questions above.

**Were the speakers determined by the facilitator appropriately?**

The options of this questions are (a) Strongly inappropriate and unnatural, (b) Inappropriate and unnatural, (c) Felt nothing, unconscious, (d) Appropriate and natural, and (e) Strongly appropriate and natural. Table 6 shows the results. By using the proposed method, the percentage of respondents who answered “Inappropriate” (includes (a) and (b)) increased from 9% to 33%, compared to the baseline method. Even though the proposed method had a good effect, the number of people who felt that the method was inappropriate or unnatural was larger for the proposed method than that for the baseline. One of the reasons for this result was that few participants answered that the number of speakers selected seemed to be either too many and frequent, or too few in the proposed method. Since the baseline method selected the next speaker randomly, the total number of selected participants could have been evenly distributed. However, the proposed method could be considered to be somewhat “picky” and the total number of selected participants could have been uneven.

**Table 6.** Was the speaker determination by the facilitator appropriate?

	(a)	(b)	(c)	(d)	(e)
Baseline Method	0 (0.0%)	2 (9.5%)	14 (66.7%)	4 (19.0%)	1 (4.8%)
Proposed Method	0 (0.0%)	7 (33.3%)	9 (42.9%)	5 (23.8%)	0 (0.0%)

**Evaluate the other participants’ discussion abilities.** The options of this questions are (a) Very poor, (b) Poor, (c) Neither high nor poor, (d) High, and (e) Very high. We defined “Very poor” as 0 and “Very high” as 5 and quantified each participant’s discussion ability by adding the total amount

of answers. To confirm the validity, we calculated ① The correlation coefficient  $\rho_1$  between “the discussion abilities” and “the average of the discussion score” of each subject, and ② The correlation coefficient  $\rho_2$  between the discussion abilities and “the total of the discussion score” of each subject. These values became  $\rho_1 = 0.037$  and  $\rho_2 = 0.349$ . Although both of them were positive values, the correlations were not very strong. Thus, improvement of the definition of the discussion scores is a possible future research option.

## 4 Conclusion

This paper proposed a method to select subsequent speakers in a discussion using the multi-armed bandit algorithm. We conducted a real discussion experiment to confirm the validity and the usefulness of this method. Based on the experimental results, we confirmed that our proposed method can positively influence discussions; for example, it selected participants who posted good ideas and opinions and prevented the discussions from having only similar ideas and opinions.

One of the possible future research options is to decide the most effective timing method when allowing someone to speak. For example, Nihei et al. [4] proposed a discussion state recognition model that recognized particular states in face-to-face Discussions. Our task is to find a way to implement the similar kind of model that can be used for online discussion. In addition, offering questions when a facilitator selects someone to speak could also be important.

**Acknowledgement.** This work was supported by JST CREST Grant Number JPMJCR15E1, Japan.

## References

1. Kotani, T., Seki, K., Matsui, T., Okamoto, T.: Development of discussion supporting system based on the “value of favorable words’ influence”. *Jpn. Soc. Artif. Intell.* **19**(2), 95–104 (2004)
2. Kuleshov, V., Precup, D.: Algorithms for the multi-armed bandit problem. *J. Mach. Learn. Res.* **1**, 1–48 (2000)
3. Nakahara, T., Ouchi, A., Uno, T., Hamuro, Y.: Extracted opinions from twitter using bipartite graph polishing. In: *The 30th Annual Conference of the Japanese Society for Artificial Intelligence*, pp. 1–4 (2016)
4. Nihei, F., Hayashi, Y., Nakano, Y.: Detecting discussion state shifts in group discussions. In: *The 28th Annual Conference of the Japanese Society for Artificial Intelligence*, pp. 1–4 (2014)
5. Ohmoto, Y., Toda, Y., Ueda, K., Nishida, T.: Analyses of the facilitating behavior by using participant’s agreement and nonverbal behavior. In: *2011 Information Processing Society of Japan*, pp. 3659–3670 (2011)
6. Ontanon, S.: Bandit algorithms in game tree search: application to computer renju. *J. Artif. Intell. Res.* **58**, 665–702 (2017)
7. Villara, S.S., Bowden, J., Wason, J.: Multi-armed bandit models for the optimal design of clinical trials: benefits and challenges. *Stat. Sci.* **30**(2), 199–215 (2015)



# An Entropy-Based Class Assignment Detection Approach for RDF Data

Molood Barati<sup>1</sup>(✉), Quan Bai<sup>1</sup>, and Qing Liu<sup>2</sup>

<sup>1</sup> Auckland University of Technology, Auckland, New Zealand  
{mbarati, quan.bai}@aut.ac.nz

<sup>2</sup> Data61, CSIRO, Sydney, Australia  
q.liu@csiro.au

**Abstract.** The RDF-style Knowledge Bases usually contain a certain level of noises known as Semantic Web data quality issues. This paper has introduced a new Semantic Web data quality issue called Incorrect Class Assignment problem that shows the incorrect assignment between instances in the instance-level and corresponding classes in an ontology. We have proposed an approach called CAD (Class Assignment Detector) to find the correctness and incorrectness of relationships between instances and classes by analyzing features of classes in an ontology. Initial experiments conducted on a dataset demonstrate the effectiveness of CAD.

**Keywords:** Semantic Web data quality issue · Ontology refinement  
Incorrect assignment · Knowledge discovery

## 1 Introduction

Recently, researchers are tackling with SW data quality issues for refining and re-engineering RDF-style Knowledge Bases (KBs). In this paper, we have identified a new SW data quality issue called Incorrect Class Assignment (ICA) problem that shows incorrect assignment between instances in the instance-level and corresponding classes in ontology. The DBpedia ontology defines a Royal class with two subclasses of BritishRoyalty and PolishKing for all royalties. There exist some instances that are incorrectly assigned to unrelated classes in ontology. For example, John I Albert, king of Poland, has been assigned to BritishRoyalty class instead of PolishKing class defined in the DBpedia ontology. This example can be described as an incorrect assignment issue between instance-level data and corresponding classes in ontology. The research problem used in this paper has been modelled in Fig. 1. We name all instances which have been correctly assigned to corresponding classes in ontology as CA. The data quality issue can be defined as the Incorrect Class Assignment problem (ICA) where at least one instance has been incorrectly assigned to class A instead of class B. Under this motivation, we proposed an approach called Class Assignment Detector (CAD) to deal with ICA problem. The main contribution of this paper is threefold



including (I) identifying and defining a new SW data quality issue called Incorrect Class Assignment problem (ICA), (II) proposing CAD approach to detect the correctness and incorrectness of relationships between instances and classes in ontology by analyzing features of classes, and (III) conducting initial experiments over DBpedia dataset 3.8 to show the effectiveness of CAD.

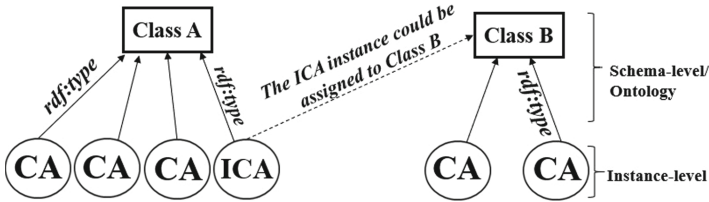


Fig. 1. Modeling the ICA problem.

This paper is structured as follows. Section 2 reviews related work on SW data quality issues. Section 3 describes the CAD approach. Section 4 reveals the experimental results. Section 5 explains the conclusion and future work.

## 2 Related Work

There exist various forms of SW data quality issues such as missing type prediction, incorrect or incomplete statements, invalid links to external resources, missing link prediction, etc. Studies based on first issue aim to predict missing *rdf:type* relation in the KBs [1, 2]. Second issue refers to the incorrect or incomplete statements of SW data [3, 4]. Consider an RDF triple (Rodrigo\_Salinas, birthPlace, Puebla.F.C.) shared by DBpedia. The DBpedia provides an invalid object value that is the name of a stadium for Rodrigo Salinas instead of sharing a city or a country name. Third common issue refers to the faulty and invalid links to external RDF-style KBs [5]. Predicting Missing links (i.e. predicates) is another frequent issue in the KBs. Consider Barack Obama as a subject and Honolulu as an object of an RDF triple. Here the question is that how to learn birthPlace relation by mining existing RDF data. A Path ranking Algorithm proposed by [6, 7] focused on this issue. To the best of our knowledge, the ICA problem has not been explicitly addressed by the most of existing work.

## 3 The Architecture of Class Assignment Detector

The CAD architecture contains two main modules including (1) Class features extraction, and (2) Instance-Class relationship analysis. In Module 1, the CAD extracts features of classes in ontology. The output of Module 1 has been used in Module 2 to assess the correctness and incorrectness of relationships between instances and classes in ontology.

### 3.1 Module 1: Class Features Extraction

Generally, a class is a category of things having some common features that make those things distinct from others. To detect the features of classes in our scenario, the initial step is to analyze instance-level data to extract common features among RDF triples. In the following, we first explain the idea behind mining common features from RDF triples. Then, we describe how mining common features of instance-level data leads CAD to extract features of classes.

**Identifying Common Features from RDF Triples.** The assertion of an RDF triple (i.e., subject, predicate, object) shows a meaningful relationship between a subject and an object provided by a predicate. Considers RDF triples (John I Albert, deathPlace, Poland) and (Casimir III, deathplace, Poland). The subjects of these RDF triples, i.e., John I Albert and Casimir III, have a common feature, i.e., (deathPlace Poland). To extract this behavior from RDF triples, we have defined the concepts of Group Feature and Common Feature as follows.

**Definition 1** (*Group Feature*). Given RDF triples, a Group Feature  $gf_i$  is a 2-tuple, i.e.,  $gf_i = (g_i, f_i)$ .  $g_i$  is a Group of subjects or objects, i.e.,  $\{s_1, s_2, \dots, s_n\}$  or  $\{o_1, o_2, \dots, o_n\}$ .  $f_i$  is a Feature shared by  $g_i$ . Corresponding with the content in  $g_i$ ,  $f_i$  contains a combination of predicate-object or predicate-subject, i.e., (p, o) or (p, s).

**Definition 2** (*Common Feature*). Given a Group Feature  $gf_i = (g_i, f_i)$ , Feature  $f_i$  is a Common Feature  $cf_i$  for Group  $g_i$ , if the number of instances in the  $g_i$  is greater than or equal to the Minimum Instance Number (*MinIN*).

**Detecting Features of Classes.** RDF-style KBs suffer from ICA problem since publishing SW data is manually maintained by contributors. To this end, CAD takes advantage from information theory [8] to analyze the level of uncertainty from this situation. As explained, a Common Feature shows a common behavior of instances (subjects or objects) in a Group. The information gain allows us to measure which common features are more certain to be used as features of classes in ontology. The following first introduces a measure based on entropy that calculates the uncertainty associated with the whole random space. In the SW, we have defined the entropy of a random space as follows.

**Definition 3** (*Random Space Entropy*). Given an ontology, a Random Space  $S$  is a space built up from instances of different classes in ontology. The entropy of  $S$  can be calculated by Eq. 1:

$$Entropy(S) = - \sum_{i=1}^N p(c_i) \log_2 p(c_i) \quad (1)$$

where  $N$  is the total number of classes in the ontology and  $p(c_i) = \frac{|Ins_{c_i}|}{|INS|}$  is the probability of Class  $c_i$  in the Random Space  $S$ .  $|Ins_{c_i}|$  is the total number

of instances of Class  $c_i$ .  $|INS|$  is the total number of instances in the Random Space  $S$ .

In the information theory, more information can be obtained by a random space with lower entropy, and vice versa. In our scenario, a Common Feature can be shared by instances of different classes in ontology. Therefore, the information gained from a Common Feature depends on the types of instances in its Group. Few types can cause lower entropy and consequently more information gained from the Common Feature. By relying on the fact, we have defined the concepts of Common Feature Information and Common Feature Space as follows.

**Definition 4** (*Common Feature Information*). Given a Random Space  $S$  and a Common Feature  $cf_i$ , the information gained from  $cf_i$  is a normalized value measured by Eq. 2:

$$NormGain(S, cf_i) = \frac{Entropy(S) - \frac{|Ins_{cf_i}|}{|INS|} Entropy(S_{cf_i})}{Entropy(S)} \quad (2)$$

where  $Entropy(S) \neq 0$  and  $0 \leq NormGain(S, cf_i) \leq 1$ .

**Definition 5** (*Common Feature Space*). Given a Common Feature  $cf_i$ , RDF triples and its corresponding ontology, a Common Feature Space  $S_{cf_i}$  is a space built up from instances that share the Common Feature  $cf_i$ . The Entropy of Common Feature Space  $S_{cf_i}$  can be computed by Eq. 3:

$$Entropy(S_{cf_i}) = - \sum_{i=1}^n p(c_{i.cf_i}) \log_2 p(c_{i.cf_i}) \quad (3)$$

where  $n$  is the total number of classes in the Common Feature Space  $S_{cf_i}$  and  $p(c_{i.cf_i}) = \frac{|Ins_{c_i.cf_i}|}{|Ins_{cf_i}|}$  is the probability of class  $c_i$  in the Common Feature Space  $S_{cf_i}$ .  $|Ins_{c_i.cf_i}|$  is the total number of instances of class  $c_i$  that share  $cf_i$ .  $|Ins_{cf_i}|$  is the total number of instances that share  $cf_i$ .

According to Eq. 2, the more information gained by a Common Feature indicates fewer types in the random space generated by the Common Feature.

On the one side, a Common Feature can be shared by instances of different classes. On the other side, instances of a class might share more than one Common Feature. In this paper, we have evaluated the information gained from combinations of common features. Consider Fig. 2 to explain the idea. Common Feature  $cf_1$  is shared by instance  $ins_1$  with *rdf:type D*, and  $ins_2, ins_i, ins_n$  with *rdf:type A*, and  $ins_j$  with *rdf:type B*. Common Feature  $cf_2$  is also shared by  $ins_2, ins_i, ins_n$  with *rdf:type A*, and  $ins_k$  with *rdf:type C*, and  $ins_m$  with *rdf:type E*. The combination of  $(cf_1, cf_2)$  gains more information to compare with each  $cf_1$  and  $cf_2$ . Because the Random Space of  $(cf_1, cf_2)$  contains lower entropy and fewer types (i.e., types *A* and *B*) to compare with  $cf_1$  and  $cf_2$  that contain instances with  $\{rdf:type A, rdf:type B, rdf:type D\}$  and  $\{rdf:type A, rdf:type C, rdf:type E\}$ , respectively. Based on this motivation, the concept of Virtual Common Feature is defined as follows.

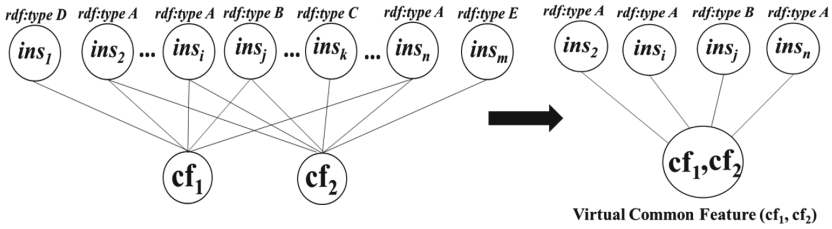


Fig. 2. An example of a Virtual Common Feature.

**Definition 6** (*Virtual Common Feature*). A Virtual Common Feature  $vcf$  is a combination of  $n$  ( $n \geq 2$ ) common features where the number of instances that share  $vcf$  is greater than or equal to  $MinIN$ .

A Virtual Common Feature  $vcf$  shows there is a number of instances that share more than one Common Feature. In this regard, the information gained from a Virtual Common Feature can be computed by Eqs. 2 and 3. We just need to replace  $|Ins_{cf_i}|$  and Entropy ( $S_{cf_i}$ ) with  $|Ins_{vcf_i}|$  and Entropy ( $VS_{vcf_i}$ ). The  $|Ins_{vcf_i}|$  is the total number of instances that share  $vcf_i$  and Entropy ( $VS_{vcf_i}$ ) is the Entropy of Virtual Common Feature Space.

Given a Common Feature  $cf_i$  and a Virtual Common Feature  $vcf_i$ , the more information indicates lower entropy (i.e., lower uncertainty) in the random spaces generated by  $cf_i$  and  $vcf_i$ . This fact reveals that most of instances that have shared  $cf_i$  and  $vcf_i$  have the same type. Based on this motivation, the concept of Class Feature is defined as follows.

**Definition 7** (*Class Feature*). A Common Feature  $cf_i$  or a Virtual Common Feature  $vcf_i$  is a Class Feature for Class  $c_i$ , if the information gained from  $cf_i$  or  $vcf_i$  is greater than or equal to NormGain Thresholds ( $NGTh$ ).

Note that  $cf_i$  (or  $vcf_i$ ) is a Class Feature for Class  $c_i$  where most instances that share  $cf_i$  have been assigned to Class  $c_i$ . It is important to mention that a class can have multiple class features including common features and virtual common features with information gained greater than or equal to  $NGTh$ .

### 3.2 Module 2: Instance-Class Relationship Analysis

If we take an instance with type and features, the goal of Module 2 is to analyse the correctness (i.e., CA) and incorrectness (i.e., ICA) of relationships between the instance and classes. To this end, Algorithm 1 has been implemented to assess the above targets for a given instance in four different statuses including (I) the CA status of an instance with one Feature, (II) the ICA status of an instance with one Feature, (III) the CA status of an instance with multiple features, and (IV) the ICA status of an instance with multiple features. Algorithm 1 receives Minimum Instance Number ( $MinIN$ ), NormGain Thresholds ( $NGTh$ ), classes ( $C$ ), features of classes ( $C.features$ ), instances ( $Iset$ ), and features of

instances ( $I.features$ ) as inputs. Algorithm 1 returns  $CA$  and  $ICA$  statuses for given instances as an output. Note that the status of  $ins_i$  is *Undecidable* if  $ins_i$  is neither  $CA$  nor  $ICA$ .

---

**Algorithm 1.** Instance-Class relationship
 

---

```

input   :  $MinIN, NGTh, C, C.features, Iset, I.features$ 
output  :  $CA$  and  $ICA$ 
1   $ICA \leftarrow \emptyset, CA \leftarrow \emptyset, Undecidable \leftarrow \emptyset$ ;
2  for each  $ins_i \in Iset$  do
3    if  $ins_i$  shares one Feature then
4      IF the feature of  $ins_i$  is a Common Feature, then OneCommonFeature
      flag will be true; ELSE  $ins_i$  is neither  $CA$  nor  $ICA$  and it will record in
      Undecidable set, then the algorithm iterates for another instance;
5    else
6      1. The features of  $ins_i$  will check and those which are common features will
      record in  $CF_{ins_i}$ ;
7      2. IF  $CF_{ins_i}$  has one Common Feature, then OneCommonFeature flag
      will be true; ELSEIF  $CF_{ins_i}$  has no Common Feature, so  $ins_i$  is neither
       $CA$  nor  $ICA$  and will record in Undecidable set, then the algorithm iterates
      for another instance; ELSE all common features of  $CF_{ins_i}$  will store in
       $Feature_{ins_i}$  set;
8      3. IF OneCommonFeature is not true, then the algorithm checks if
       $Feature_{ins_i} \geq MinIN$ , if yes, then Virtual Common Feature will create
      with  $vcf_{ins_i} = Feature_{ins_i}$  and Multiplecommonfeatures flag will be
      true; ELSE  $ins_i$  is neither  $CA$  nor  $ICA$  and will record in Undecidable set,
      then the algorithm iterates for another instance;
9    if (OneCommonFeature) then
10     IF Information gained by Common Feature of  $ins_i \geq NGTh$ , then do 4 and
     5; ELSE  $ins_i$  is neither  $CA$  nor  $ICA$  and it will record in Undecidable set,
     then the algorithm iterates for another instance;
11     4. The common feature of  $ins_i$  will check in the features of Class  $c_j \in C$ .;
12     5. IF the feature of  $ins_i$  is in the class features of  $c_j$  and if  $ins_i$  has the same
     type with class  $c_j$ , then  $ins_i$  will record in  $CA$ .; ELSE  $ins_i$  has an  $ICA$ 
     status.;
13   if (Multiplecommonfeatures) then
14     IF Information gained by Virtual Common Feature  $vcf_{ins_i} \geq NGTh$ , then
     do 6; ELSE  $ins_i$  is neither  $CA$  nor  $ICA$  and it will record in Undecidable
     set, then the algorithm iterates for another instance;
15     7. IF the type of  $ins_i$  is equal to the classType of Virtual Common Feature
      $vcf_{ins_i}$ , then  $ins_i$  will record in  $CA$ ; ELSE  $ins_i$  has an  $ICA$  status.;
16 return  $CA$  and  $ICA$ 

```

---

## 4 Experiments and Analysis

The following experiments have been conducted over DBpedia dataset 3.8 that is one of the most common errors encountered RDF-style KBs. By using DBpedia dataset, we considered two classes called Food and Hotel. Each class contains about 750 instances. The goal of following experiments is to check the accuracy of CAD in analyzing the correctness and incorrectness of relationships between instances and classes. Generally, the accuracy of a system is the degree of closeness between a measured value and the true value. In our scenario, a measured value refers to the number of correctly (i.e.,  $CA$ ) and incorrectly (i.e.,  $ICA$ )

assigned instances detected by CAD approach. While a true value indicates the predefined number of CA and ICA instances in a class. Thus, we correctly assigned 700 instances to the Hotel class that indicates a true value for CA instances. We also incorrectly assigned 50 instances of Hotel class to the Food class that shows a true value for ICA instances. To measure the accuracy of CAD approach, two measurements called  $Accuracy_{CA}$  and  $Accuracy_{ICA}$  are defined as follows. Given a class, the accuracy of CAD in detecting correctly assigned instances can be computed by Eq. 4:

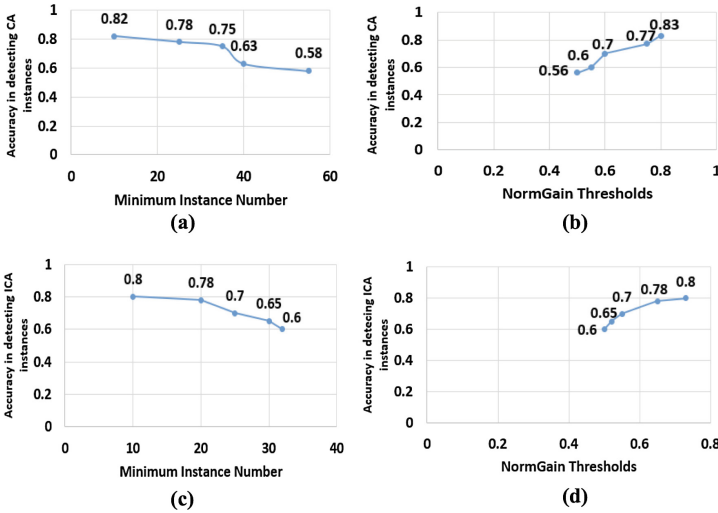
$$Accuracy_{CA} = \frac{|ins_{CA} \cap INS_{CA}|}{|INS_{CA}|} \tag{4}$$

where  $ins_{CA}$  is the number of correctly assigned instances detected by CAD and  $INS_{CA}$  is a true value for the predefined number of CA instances in the class.

Given a class, the accuracy of CAD in detecting incorrectly assigned instances can be measured by Eq. 5:

$$Accuracy_{ICA} = \frac{|ins_{ICA} \cap INS_{ICA}|}{|INS_{ICA}|} \tag{5}$$

where  $ins_{ICA}$  is the number of incorrectly assigned instances detected by CAD and  $INS_{ICA}$  is a true value for the predefined number of ICA instances in the class.



**Fig. 3.** (a) Accuracy in detecting CA instances in different  $MinIN$ , (b) Accuracy in detecting CA instances in different  $NGTh$ , (c) Accuracy in detecting ICA instances in different  $MinIN$ , (d) Accuracy in detecting ICA instances in different  $NGTh$ .

Figure 3(a) shows that the accuracy of CAD approach in detecting CA instances has been gradually decreased by increasing  $MinIN$ . One reason behind

such reduction is related to the strategy of selecting common features by using *MinIN*. Consider the process of analyzing common features in Algorithm 1. Given an instance, if the feature shared by  $ins_i$  is not a Common Feature, then the status of  $ins_i$  is undecidable. For example, an RDF triple (White House, location, Herm) detected indicates that White House is an instance of Hotel class with a particular feature i.e., (location, Herm). In the Hotel class, Herm is the only instance that has shared (location, Herm) as a particular feature. Algorithm 1 ignores some instances in case the features shared by them have not identified as common features. Figure 3(b) shows that the accuracy of CAD approach in detecting CA instances has been grown by increasing *NGTh*. Figure 3(c) represents that the accuracy of CAD in detecting ICA instances is reduced by increasing *MinI*. Consider again the RDF triple (White House, location, Herm). If White House has been incorrectly assigned to the Food class, Algorithm 1 ignores White House since (location, Herm) has not identified as a Common Feature.

## 5 Conclusions and Future Work

This paper has introduced a new SW data quality issue called Incorrect Class Assignment (ICA) problem that indicates incorrect assignment between instance-level data and corresponding classes in an ontology. So, we proposed an entropy-based approach called Correct Assignment Detector (CAD) to deal with ICA problem. A direction for future work is to apply Natural Language Processing (NLP) techniques on predicates of common features to find out more similar behaviors taken by instances in the groups.

## References

1. Melo, A., Völker, J., Paulheim, H.: Type prediction in noisy RDF knowledge bases using hierarchical multilabel classification with graph and latent features. *Int. J. Artif. Intell. Tools* **26**(02), 1760011 (2017)
2. Gunaratna, K., Thirunarayan, K., Sheth, A., Cheng, G.: Gleaning types for literals in RDF triples with application to entity summarization. In: Sack, H., Blomqvist, E., d'Aquin, M., Ghidini, C., Ponzetto, S.P., Lange, C. (eds.) *ESWC 2016*. LNCS, vol. 9678, pp. 85–100. Springer, Cham (2016). [https://doi.org/10.1007/978-3-319-34129-3\\_6](https://doi.org/10.1007/978-3-319-34129-3_6)
3. Lehmann, J., Gerber, D., Morsey, M., Ngonga Ngomo, A.-C.N.: DeFacto - deep fact validation. In: Cudré-Mauroux, P., Heflin, J., Sirin, E., Tudorache, T., Euzenat, J., Hauswirth, M., Parreira, J.X., Hendler, J., Schreiber, G., Bernstein, A., Blomqvist, E. (eds.) *ISWC 2012*. LNCS, vol. 7649, pp. 312–327. Springer, Heidelberg (2012). [https://doi.org/10.1007/978-3-642-35176-1\\_20](https://doi.org/10.1007/978-3-642-35176-1_20)
4. Töpper, G., Knuth, M., Sack, H.: DBpedia ontology enrichment for inconsistency detection. In: *Proceedings of the 8th International Conference on Semantic Systems*, pp. 33–40. ACM (2012)

5. Volz, J., Bizer, C., Gaedke, M., Kobilarov, G.: Discovering and maintaining links on the web of data. In: Bernstein, A., Karger, D.R., Heath, T., Feigenbaum, L., Maynard, D., Motta, E., Thirunarayan, K. (eds.) ISWC 2009. LNCS, vol. 5823, pp. 650–665. Springer, Heidelberg (2009). [https://doi.org/10.1007/978-3-642-04930-9\\_41](https://doi.org/10.1007/978-3-642-04930-9_41)
6. Lao, N., Cohen, W.W.: Relational retrieval using a combination of path constrained random walks. *Mach. Learn.* **81**(1), 53–67 (2010)
7. Lao, N., Mitchell, T., Cohen, W.W.: Random walk inference and learning in a large scale knowledge base. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 529–539. Association for Computational Linguistics (2011)
8. Jaynes, E.T.: Information theory and statistical mechanics. *Phys. Rev.* **106**(4), 620 (1957)





# Weighted Double Deep Multiagent Reinforcement Learning in Stochastic Cooperative Environments

Yan Zheng<sup>1</sup>, Zhaopeng Meng<sup>1,2</sup>, Jianye Hao<sup>1(✉)</sup>, and Zongzhang Zhang<sup>3</sup>

<sup>1</sup> Tianjin University, Tianjin, China  
jianye.hao@tju.edu.cn

<sup>2</sup> Tianjin University of Traditional Chinese Medicine, Tianjin, China

<sup>3</sup> Soochow University, Suzhou, China

**Abstract.** Recently, multiagent deep reinforcement learning (DRL) has received increasingly wide attention. Existing multiagent DRL algorithms are inefficient when faced with the non-stationarity due to agents update their policies simultaneously in stochastic cooperative environments. This paper extends the recently proposed weighted double estimator to the multiagent domain and propose a multiagent DRL framework, named weighted double deep Q-network (WDDQN). By utilizing the weighted double estimator and the deep neural network, WDDQN can not only reduce the bias effectively but also be extended to scenarios with raw visual inputs. To achieve efficient cooperation in the multiagent domain, we introduce the lenient reward network and the scheduled replay strategy. Experiments show that WDDQN outperforms the existing DRL and multiagent DRL algorithms, i.e., double DQN and lenient Q-learning, in terms of the average reward and the convergence rate in stochastic cooperative environments.

**Keywords:** Multiagent learning · Deep learning · Weighted Q-learning

## 1 Introduction

The goal of reinforcement learning (RL) is to learn an optimal behavior within an unknown dynamic environment, usually modeled as a Markov decision process (MDP), through trial and error [11]. By far, deep RL (DRL) has achieved great successes in mastering various complex problems [6], which can be credited to the *experience replay* and *target network* [6, 10].

In multiagent domains, approaches like [5, 13] have been proposed by extending Q-learning to address the coordination problems in cooperative multiagent systems (MAS). However, they are only verified using relative simple problems. Recently, employing DRL in MAS draws wide attention [3, 4, 7]. These algorithms, however, still suffer from two intrinsic difficulties: stochasticity due to the noisy reward signals; and non-stationarity due to the dynamicity of coexisting agents. The stochasticity introduces additional biases in estimation,

while the non-stationarity harms the effectiveness of experience replay, which is crucial for stabilizing deep Q-networks. These two characteristics result in the lack of theoretical convergence guarantees and amplify the difficulty of finding the optimal Nash equilibriums, especially in cooperative multiagent problems.

This work focuses on learning algorithms of independent learners (ILs) in cooperative MAS. In such setting, agents are unable to observe other agents' actions and rewards [2], share a common reward function and learn to maximize the common expected discounted reward (a.k.a. return). To handle the stochastic and non-stationary challenges in MAS, we propose the weighted double deep Q-network (WDDQN) with two auxiliary mechanisms, the *lenient reward network* and the *scheduled replay strategy*, to help ILs in finding the optimal policy, maximizing the common return.

Our contributions are three-fold. First, we extend weighted double Q-learning (WDQ) [14], a state-of-the-art traditional RL method, to the multiagent DRL settings. Second, we introduce lenient reward network inspired by the lenient Q-learning [7, 8]. Third, we propose a scheduled replay strategy to stabilize and speed up the learning process in complex multiagent problems with raw visual inputs. Empirical results demonstrate that on a fully cooperative multiagent problem, WDDQN with new mechanisms contribute to increasing the algorithm's convergence, decreasing the instability and helping ILs to find an optimal policy simultaneously.

## 2 Preliminaries

**Weighted Double Q-Learning (WDQ).** [14] uses a dynamic heuristic value  $\beta$  to balance between the overestimation of the single estimator and the underestimation of the double estimator during the iterative Q-value update process:

$$Q(s, a)^{U, WDQ} = \beta Q^U(s, a^*) + (1 - \beta) Q^V(s, a^*), \quad (1)$$

where a linear combination of two estimators  $Q^U$  and  $Q^V$  is used for updating Q-value. When  $a^*$  is chosen by  $Q^U$ , i.e.,  $a^* \in \arg \max_a Q^U(s, a)$ ,  $Q^U(s, a^*)$  will be positively biased and  $Q^V(s, a^*)$  will be negatively biased, and vice versa.  $\beta \in [0, 1]$  balances between the positive and negative biases.

**Lenient Q-Learning.** [9] updates the policies of multiple agents towards an optimal joint policy simultaneously by letting each agent adopt an optimistic dispose at the initial exploration phase. This contributes to discovery the optimal joint policy and has been empirically verified in [7, 8, 13]. To be specific, during training, lenient agents keep track of the temperature  $T_t(s, a)$  for each state-action pair  $(s, a)$  at time  $t$ , which is initially set to a defined maximum temperature value and used for measuring the leniency  $l(s, a)$  as follows:

$$l(s_t; a_t) = 1 - e^{-K * T_t(s_t, a_t)}, \quad (2)$$

where  $K$  is a constant determining how the temperature affects the decay in leniency. As suggested by [13],  $T_t(s_t, a_t)$  is decayed using a discount factor  $\kappa \in [0, 1]$  and  $T_{t+1}(s_t, a_t) = \kappa T_t(s_t, a_t)$ . Given the TD error  $\delta = Y_t^Q - Q_t(s_t, a_t; \theta_t)$ , the iterative update formula of lenient Q-learning is defined as follows:

$$Q(s_t, a_t) = \begin{cases} Q(s_t, a_t) + \alpha \delta & \text{if } \delta > 0 \text{ or } x > l(s_t, a_t), \\ Q(s_t, a_t) & \text{otherwise.} \end{cases} \quad (3)$$

The random variable  $x \sim U(0, 1)$  is used to ensure that a negative update  $\delta$  is performed with a probability  $1 - l(s_t, a_t)$ . We absorb this interesting notion of forgiveness into our lenient reward network to boost the convergence in cooperative Markov games which will be explained later.

### 3 Weighted Double Deep Q-Networks

In the section, we introduce a new multiagent DRL algorithm, weighted double deep Q-networks (WDDQN), with two auxiliary mechanisms, i.e., the lenient reward approximation and the scheduled replay strategy, to achieve efficient coordination in stochastic multiagent environments, where reward could be extremely stochastic due to the environments' inherent characteristics and the continuous change of the coexisting agents' behaviors. For the stochastic environments, WDDQN uses the combination of the weighted double estimator and a reward approximator to reduce the estimation error. As for the non-stationary coexisting agents, we incorporate the notion of leniency [7, 8] into the reward approximator to provide an optimistic estimation of the expected reward under each state-action pair  $r(s, a)$ . Besides, directly applying prioritized experience replay [10] in multiagent DRL leads to poor performance, as stored transitions can become outdated because agents update their policies simultaneously. To address this, we propose a scheduled replay strategy to enhance the benefit of prioritization by adjusting the priority for transition sample dynamically. In the remainder of this section, we will describe these facets in details.

**WDDQN** is adapted from WDQ by leveraging the neural network as the Q-value approximator to handle problems with high-dimensional state spaces. The network architecture is depicted in Fig. 1. To reduce the estimation bias, the combination of two estimators, represented as Deep Q-networks  $Q^U$  and  $Q^V$  with the same architecture, is adopted to select action  $a = \max_{a'} \frac{Q^U(s, a') + Q^V(s, a')}{2}$ . Besides, the target  $Q^{\text{Target}}(s, a)$  used for Q-value updating in back-propagation is replaced with a weighted combination like Eq. 1, balancing between the overestimation and underestimation. Besides, a lenient reward approximator and an efficient scheduled replay strategy are incorporated in WDDQN to achieve bias reduction and efficient coordination in multiagent stochastic environments.

**Lenient Reward Network (LRN)** is a neural network estimator to explicitly approximate the reward function  $R(s, a)$ , being conducive to reduce noise in

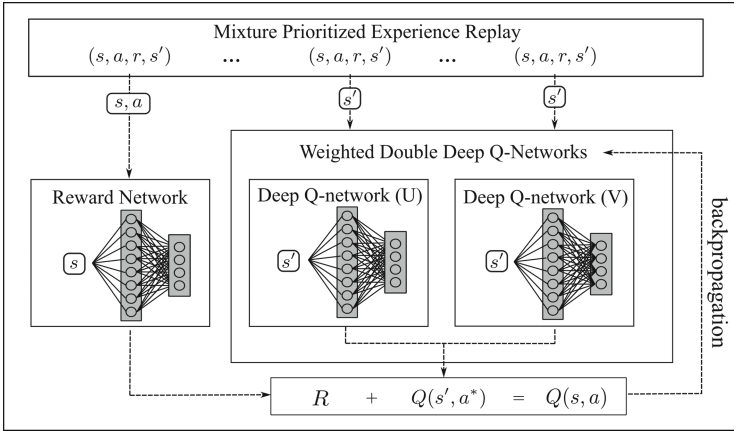


Fig. 1. Network architecture of WDDQN

stochastic rewards. LRN can reduce bias in immediate reward  $r$  yielded from stochastic environments by averaging all rewards for distinct  $(s, a)$  pair and be trained using the transitions stored in the experience replay during the online interaction. Instead of using the reward  $r$  in transition  $(s, a, r, s')$  from experience memory, WDDQN uses the estimated reward by the reward network as shown in Fig. 1.

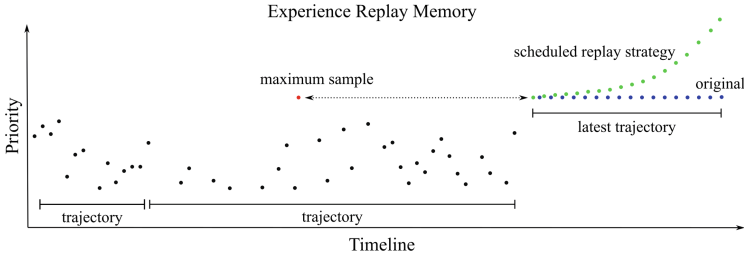
In addition to stochasticity, in a cooperative MAS, the coexisting agents introduce additional bias to  $r$  as well. The mis-coordination may lower the reward  $r$  for  $(s, a^*)$  although the agent has adopted the optimal action. To address this, LRN draws on the lenient concept in [9], making the agent keep optimistic during the initial exploration phase. LRN is updated periodically as follows:

$$R_{t+1}(s_t, a_t) = \begin{cases} R_t(s_t, a_t) + \alpha\delta & \text{if } \delta > 0 \text{ or } x < l(s_t, a_t), \\ R_t(s_t, a_t) & \text{otherwise.} \end{cases} \quad (4)$$

where  $R_t(s_t, a_t)$  is the reward approximation of state  $s$  and action  $a$  at time  $t$ , and  $\delta = \bar{r}_t^{(s,a)} - R_t(s_t, a_t)$  is the TD error between the  $R_t(s_t, a_t)$  and the target reward  $\bar{r}_t^{(s,a)} = 1/n \sum_{i=1 \dots n} r_i^{(s,a)}$  obtained by averaging all immediate reward  $r_i^{(s,a)}$  of  $(s, a)$  pairs in experience memory. Note that  $l(s_t, a_t)$  inherits from Eq. 3, and gradually decayed each time a state-action  $(s, a)$  pair is visited. Consequently, the LRN contributes to reduce bias by reward approximation and can help agents to find optimal joint policies in cooperative Markov games.

**Scheduled Replay Strategy (SRS)** is a new strategy adapted from the prioritized experience replay (PER), selecting vital samples to replay in stochastic multiagent environments. In vanilla PER, the probability of a sample being chosen for training is proportional to its TD error. However, in stochastic multiagent environments, due to the noisy reward and the continuous behavior changes of

coexisting agents, the vanilla PER may deteriorate the algorithm’s convergence and perform poorly. Given a transition  $(s, a, r, s, d)$  with an extremely biased reward  $r$ , PER will treat it as an important sample for its large TD error and will frequently select it for update the network, though it is incorrect due to the big noise in  $r$ . To address this, we replace  $r$  with an estimation  $R^N(s, a)$  using LRN to correct TD error, by which the PER can distinguish true important samples.



**Fig. 2.** Comparison between the prioritized experience replay and the scheduled replay strategy: each dot represents a sample  $(s, a, r, s)$ , and a trajectory consists of an ordered sequence of samples. The x-axis represents the order that each sample comes into the replay memory and the y-axis is the priority of each sample. (Color figure online)

Another potential problem is that PER gives all samples in the new trajectory the same priority, thus resulting in the indistinguishability of importance for all new samples. To be specific, in Fig. 2, the sample with the maximum priority is colored by red dot. PER gives all samples (blue dots) in the latest trajectory with an identical priority<sup>1</sup>. However, in cooperative multiagent environments, the trajectories that agents succeed in cooperation are relatively rare, and in these trajectories, the samples closer to the terminal state is even more valuable than the ones far from the terminal state. Besides, the  $Q(s, a) = r + Q(s', a^*)$  far from the terminal state can further deteriorate if bootstrap of action value  $Q(s', a^*)$  is already highly inaccurate, since inaccurate estimation will propagate throughout the whole contiguous samples.

These two traits explain why samples close to the terminal state should be frequently used for network training. To this end, we propose the scheduled replay strategy, using a precomputed rising schedule  $[w_0, w_1, \dots, w_n]$  with size  $n$  to assign different priorities according to the sample’s position  $i$  in the trajectory. The values for  $w_i = e^{\rho c * u^i}$  are computed using an exponent  $\rho^c$  which grows with a rising rate  $u > 1$  for each  $i$ ,  $0 \leq i < n$ . The priority  $p_i$  assigned to sample with index  $i$  is obtained by multiplying the current maximum priority  $p_{\max}$  in experience memory (priority of the red point in Fig. 2) by  $w_i$ :

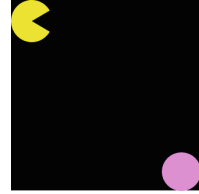
$$p_i = p_{\max} \times w_i$$

<sup>1</sup> See OpenAI source code for details: <https://github.com/openai/baselines>.

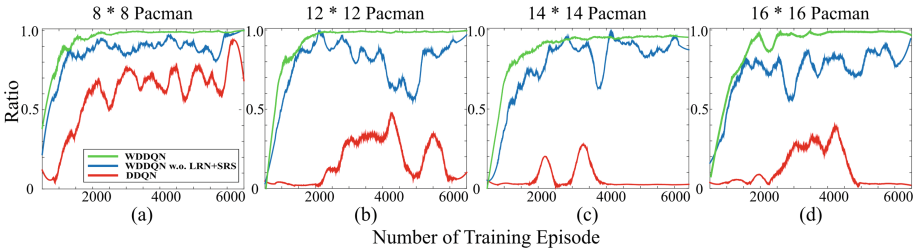
The SRS assigns a higher priority to samples near the terminal state (the green dot in Fig. 2) to ensure they are more likely to be sampled for network training. In this way, the estimation bias of the  $Q(s, a)$  near the terminal state is expected to decrease rapidly. This can significantly speed up the convergence and improve the training performance, as to be verified in the following section.

## 4 Experiments

**Pacman Game** is an  $n \times n$  gridworld problem (Fig. 3), where the agent starts at the  $s_0$  (top left cell) and moves towards the goal (pink dot) using four actions: {north, south, east, west}. Every action leads the agent to move one cell in the corresponding direction, and the goal appears randomly in any position. A stochastic reward of  $-30$  or  $40$  is received with equal probability for any action entering into the goal. Moving north or west will get a reward of  $-10$  or  $+6$ , and south or east get  $-8$  or  $+6$  at a non-goal state. The environment is extremely noisy due to the uncertainty in the reward function.



**Fig. 3.** Pacman game (Color figure online)



**Fig. 4.** Comparisons of double QN (DDQN), WDDQN with/without LRN and SRS, denoted by WDDQN and WDDQN w.o. LRN+SRS, on Pacman with 4 different sizes. The X-axis is the number of training episodes and the Y-axis is a ratio of the number of minimum steps to the goal to the number of steps that the agent actually used during training.

As shown in Fig. 4, under extremely stochastic environments, DDQN takes a long time to optimize the policy, while WDDQN and WDDQN w.o. LRN+SRS learn fast due to the weighted double estimator. DDQN and WDDQN w.o. LRN+SRS oscillate too frequently to converge, while WDDQN performs steadily and smoothly due to LRN. Another finding is that the training speed of WDDQN is faster than the others, which is attributed to the SRS. In general, WDDQN works not as well as in relatively simple RL problems and both DDQN and WDDQN w.o. LRN+SRS may not converge even after a very long training time. By contrast, WDDQN learns efficiently and steadily due to the LRN and SRS.

**Predator Game** is a more complex cooperative problem adapted from [1], where two agents (robots) try to enter into the goal state (G or S) at the same time to achieve coordination. S is a suboptimal goal with +10 reward while G is a global optimal with +80 reward. A thick wall (in gray) in the middle separates the area into two zones. Different from settings in Pacman, a reward of 0 is received whenever entering into a non-goal state and a reward of  $-1$  is received as punishment for miscoordination. We investigate whether WDDQN and related algorithms can find cooperative policies moving towards the S (suboptimal) or G (optimal), especially the optimal policy (Fig. 5).

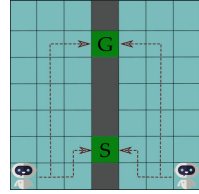


Fig. 5. Predator game

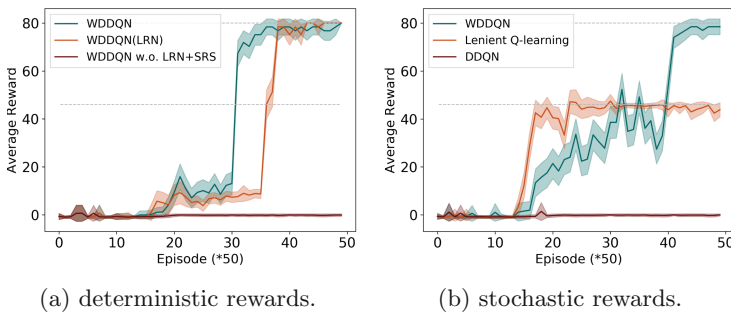


Fig. 6. (Left) Comparisons of WDDQN and its variants using the Predator game with deterministic rewards; and (right) comparisons of WDDQN and other algorithms using the Predator game with stochastic rewards. Note that, each point in the x-axis consists of 50 episodes, the y-axis is the corresponding averaged reward, and the shadow area ranges from the lowest to the highest rewards.

**Evaluation on WDDQN.** Comparison of WDDQN, WDDQN(LRN), which uses only LRN, and WDDQN w.o. LRN+SRS in terms of the average reward is conducted and depicted in Fig. 6(a). As the learning convergence is no longer guaranteed in multiagent domains, WDDQN w.o. LRN+SRS’s thus fails in finding the cooperative policy by directly combining WDQ with the neural network. By contrast, WDDQN(LRN), due to the LRN, achieves coordination more quickly and efficiently finds the optimal policy. Moreover, by leveraging the SRS, WDDQN learns the optimal policy much faster than the two others.

**Evaluation Against Other Algorithms.** Comparisons of WDDQN, DDQN [12] and lenient Q-learning [7] is conducted under the same settings except that the agent receives a reward of +10 or +100 with the possibility of 60% or 40% at goal S and a deterministic reward of +80 at goal G. S is still suboptimal as its average reward is 46. This stochasticity may mislead the agent to converge

to the suboptimal goal where a higher reward may appear accidentally. Results in terms of the average reward are depicted in Fig. 6(b), where two dashed lines indicate optimal (80) and suboptimal (46) solutions. Both WDDQN and lenient Q-learning outperform DDQN in terms of the convergence speed and the average reward in all experiments, which confirms the infeasibility of directly applying DRL algorithms in multiagent problems. Note that, WDDQN, due to the LRN and SRS, is more stable, performs better and is more likely to find the optimal solution than lenient Q-learning in such a stochastic environment.

## 5 Conclusion

We propose WDDQN with the lenient reward network and the scheduled replay strategy to boost the training efficiency, stability and convergence under stochastic multiagent environments with raw image inputs, stochastic rewards, and large state spaces. Empirically, WDDQN achieves promising performance in terms of the average reward and convergence rate on two stochastic environments.

**Acknowledgments.** The work is supported by the National Natural Science Foundation of China under Grant No.: 61702362, Special Program of Artificial Intelligence of Tianjin Municipal Science and Technology Commission (No.: 569 17ZXRGGX00150) and Science and Technology Program of Tianjin, China (Grant Nos. 15PTCYSY00030 and 16ZXHLGX00170).

## References

1. Benda, M., Jagannathan, V., Dodhiawala, R.: On optimal cooperation of knowledge sources - an empirical investigation. Technical report BCS-G2010-28, Boeing Advanced Technology Center, Boeing Computing Services (1986)
2. Claus, C., Boutilier, C.: The dynamics of reinforcement learning in cooperative multiagent systems. In: AAAI Conference on Artificial Intelligence (AAAI), pp. 746–752 (1998)
3. Gupta, J.K., Egorov, M., Kochenderfer, M.: Cooperative multi-agent control using deep reinforcement learning. In: Sukthankar, G., Rodriguez-Aguilar, J.A. (eds.) AAMAS 2017. LNCS (LNAI), vol. 10642, pp. 66–83. Springer, Cham (2017). [https://doi.org/10.1007/978-3-319-71682-4\\_5](https://doi.org/10.1007/978-3-319-71682-4_5)
4. Lanctot, M., Zambaldi, V., Gruslys, A., Lazaridou, A., Perolat, J., Silver, D., Graepel, T., et al.: A unified game-theoretic approach to multiagent reinforcement learning. In: Advances in Neural Information Processing Systems (NIPS), pp. 4193–4206 (2017)
5. Matignon, L., Laurent, G.J., Le Fort-Piat, N.: Independent reinforcement learners in cooperative markov games: a survey regarding coordination problems. *Knowl. Eng. Rev.* **27**(1), 1–31 (2012)
6. Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Graves, A., Riedmiller, M., Fidjeland, A.K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., Hassabis, D.: Human-level control through deep reinforcement learning. *Nature* **518**(7540), 529–533 (2015)



7. Palmer, G., Tuyls, K., Bloembergen, D., Savani, R.: Lenient multi-agent deep reinforcement learning. In: International Conference on Autonomous Agents and Multiagent Systems (AAMAS) (2018, to appear)
8. Panait, L., Sullivan, K., Luke, S.: Lenient learners in cooperative multiagent systems. In: International Conference on Autonomous Agents and Multiagent Systems (AAMAS) (2006)
9. Potter, M.A., De Jong, K.A.: A cooperative coevolutionary approach to function optimization. In: Davidor, Y., Schwefel, H.-P., Männer, R. (eds.) PPSN 1994. LNCS, vol. 866, pp. 249–257. Springer, Heidelberg (1994). [https://doi.org/10.1007/3-540-58484-6\\_269](https://doi.org/10.1007/3-540-58484-6_269)
10. Schaul, T., Quan, J., Antonoglou, I., Silver, D.: Prioritized experience replay. In: International Conference on Learning Representations (ICLR) (2016)
11. Sutton, R.S., Barto, A.G.: Reinforcement learning: An introduction. MIT press, Cambridge (1998)
12. Van Hasselt, H., Guez, A., Silver, D.: Deep reinforcement learning with double Q-learning. In: AAAI Conference on Artificial Intelligence (AAAI), pp. 2094–2100 (2016)
13. Wei, E., Luke, S.: Lenient learning in independent-learner stochastic cooperative games. *J. Mach. Learn. Res.* **17**(84), 1–42 (2016)
14. Zhang, Z., Pan, Z., Kochenderfer, M.J.: Weighted double Q-learning. In: International Joint Conference on Artificial Intelligence (IJCAI), pp. 3455–3461 (2017)



# Automatically Classifying Chinese Judgment Documents Using Character-Level Convolutional Neural Networks

Xiaosong Zhou<sup>1,2</sup>, Chuanyi Li<sup>1,2(✉)</sup>, Jidong Ge<sup>1,2(✉)</sup>, Zhongjin Li<sup>3</sup>,  
Xiaoyu Zhou<sup>1,2</sup>, and Bin Luo<sup>1,2</sup>

<sup>1</sup> State Key Laboratory for Novel Software Technology at Nanjing University, Nanjing, China

lcynju@126.com, gjdnju@163.com

<sup>2</sup> Software Institute, Nanjing University, Nanjing, China

<sup>3</sup> School of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou, China

**Abstract.** Judgment is a decision by a court or other tribunal that resolves a controversy and determines the rights and obligations of the parties. Since the establishment of the China Judgments Online System, more and more judgment documents have been stored online. With the explosive growth of the number of Chinese judgment documents, the need for automated classification methods is getting increasingly urgent. For Chinese data sets, traditional word-level methods often bring extra errors in word segmentation. In this paper, we proposed an approach based on character-level convolutional neural networks to automatically classify Chinese judgment documents. Different from traditional machine learning methods, we hand over the work of feature detection to the model. Throughout the process, the only part that requires human labor is labeling the category of each original documents. In order to prevent overfitting when the amount of training data is not very large, we use a shallow model which has only one convolution layer. The proposed approach does well in achieving high classification accuracy based on 7923 pieces of Chinese judgment documents. In the meanwhile, the effectiveness of our model is satisfactory.

**Keywords:** Chinese judgment documents · Text classification  
Character-level convolutional neural networks · Overfitting

## 1 Introduction

Judgment is a decision by a court or other tribunal that resolves a controversy and determines the rights and obligations of the parties. China Judgments Online System is a unified platform for judgment documents established by the Supreme People's Court of the People's Republic of China in 2013. There are totally over 44 million electronic judgments in this system. It takes a lot of manpower to maintain so much data. If we can divide the judgments according to the industries involved, then we can provide the judgment documents to the experts in the corresponding fields.

Text classification is an important branch of natural language processing. We can use this technology to reduce human labor to a large extent. In this paper, we propose a character-level convolutional neural networks model to automatically classify Chinese judgment documents. Our work includes: (1) propose a method to extract some parts of a judgment document which are helpful for classification, and reduce the sequence length while maintaining good performance, (2) propose a strategy for Chinese judgment documents representation, (3) apply a simple convolutional neural networks model to the judgment documents classification, aiming at training a good model without the need of too much data. To evaluate the performance of our approach, we use 7923 pieces of Chinese judgment documents which related to liabilities of product quality. We manually labeled them into 13 categories according to the statutory standard of industry division. We will prove the contributions of this paper by answering the follow three research questions:

- (1) How long is the appropriate length of a sequence of characters extracted from judgment documents? How is the performance of the classifier improved by reducing the sequence length?
- (2) How well can those judgment documents be classified by using character-level convolutional neural networks? What are the specific performances of this method?
- (3) Is it a better choice to adopt character-level convolutional neural networks compared with other methods such as word-level convolutional neural networks, recurrent neural networks, Naive Bayes, Decision Tree, Random Forest and Support Vector Machine?

The remainder of this paper is laid out as follows. Section 2 introduces related work of this paper. Section 3 introduces our approach in detail. Section 4 evaluates the proposed approach and answers the research questions and Sect. 5 concludes with a discussion of future work.

## 2 Related Work

As a classic natural language processing task, the text classification problem has been studied by a lot of scholars. In recent years, there has emerged a number of impressive results in the field of text classification. Laura processed user comments to extract the main topics mentioned as well as some sentences representative of those topics [1]. Hua built a prototype system for short text understanding which exploits semantic knowledge provided by a existing knowledge base and automatically harvested from a web corpus [2]. Maalej studied several probabilistic techniques to classify app reviews into four types that is predefined [3]. Paul proposed a matching technique for learning causal associations between features and class labels in document classification [4]. Yang studied text semantic analysis from a new angle by hypothesizing that helpfulness is an internal property of text [5]. Machine learning-based algorithms can also be applied to classify user requests in crowdsourcing requirements engineering [6]. Rousseau transformed the task of text categorization into a graph classification problem [7]. Joulin proposed a simple and efficient feature learning method for text

classification [8]. With the development of neural networks, there are more and more text classification tasks using the idea of neural networks. Kim proposed a classic convolutional neural networks (CNN) model for text classification [9]. Shang proposed a neural network-based response generator for short text conversation using the general encoder-decoder framework [10]. Ahn proposed a Neural Knowledge Model using recurrent neural networks (RNN) language model and the knowledge graph [11]. Johnson explored a sophisticated region embedding method using Long Short-Term Memory (LSTM) [12]. CNN and RNN were combined by Lai for text classification [13]. Tang introduced neural network models (ConvGRNN and LSTM-GRNN) for document level sentiment classification [14]. Zhang proposed a character-level convolutional neural networks model for understanding text [15], which is most relevant to our work. Based on the principle of simplicity and efficiency, we propose a shallow character-level convolutional neural networks model to process our data.

### 3 Approach

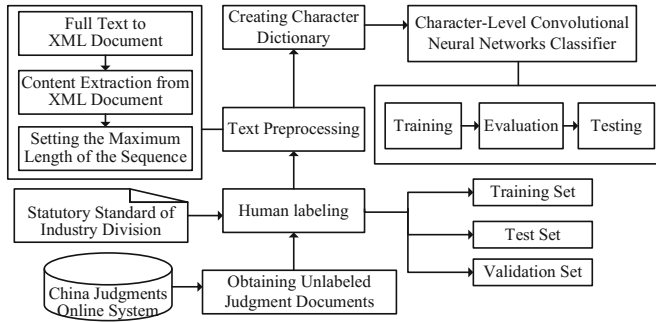
In this section, the details of our approach for Chinese judgment documents classification are described. Section 3.1 presents an overview of the framework we used in finding the approach for Chinese judgment documents. Section 3.2 introduces text preprocessing. Section 3.3 describes the model architecture and training step.

#### 3.1 Overview

As shown in Fig. 1, the classification approach starts from obtaining a lot of unlabeled judgment documents from China Judgments Online System. We obtain about 8000 pieces of judgment documents that are related to liabilities for product quality. Then we manually label those documents according to the statutory standard of industry division. We define 13 categories, which will be elaborated in Sect. 4.1. As to the data division, we adopt two strategies: (1) divide the data set into 70% of training set and 30% of test set, (2) divide the data set into 80% training set, 10% of validation set and 10% of test set. We use the `embedding_lookup` interface from tensorflow to complete the character embedding operation, which means training the character vector in the neural network.

#### 3.2 Text Preprocessing

There are generally more than 10000 characters in one Chinese judgment document. However, what can benefit our classifier is only a part of them. So we need to preprocess the text. We divide the paragraphs into seven logical blocks according to their logical relations, including “Document Head”, “Litigant”, “Judicial Records”, “Basic Information”, “Judgment Analysis Process”, “Judgment Result” and “Document tail”. What we care about is “Basic Information”. So we extract two parts of text from this block called “Plaintiff Claims” and “Facts”. For instance, we extract “Plaintiff Claims” by the following regular expression: `原告.*?诉称[\S\s]*(?=被告.*?辩称)`. During operation, each logical block is represented by one xml element, such as “Basic

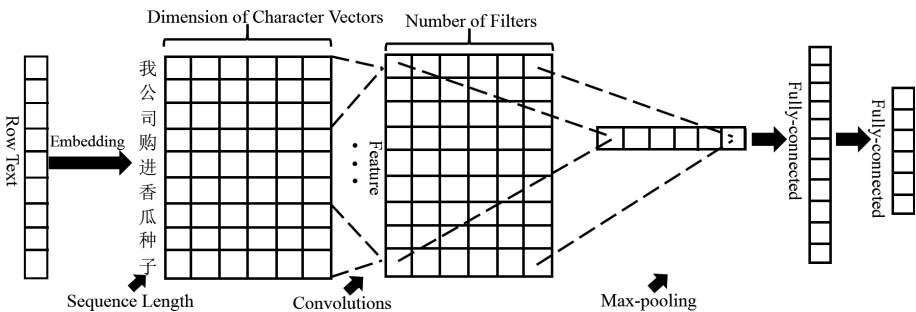


**Fig. 1.** Overview of the framework for Chinese judgment documents classification

Information”. In the meanwhile, an element can contain multiple sub-elements such as “Plaintiff Claims” and “Facts”. After doing that, the maximum length of one judgment document is reduced to 8000, which is still too long. We find that 1000 characters are enough to divide the text into 13 categories the we predefined, which will be proved in Sect. 4.2. So we simply truncate the text, leaving only the first 1000 characters.

### 3.3 Classifier

Since we use character-level convolutional neural networks as our classifier, the language itself is not important. What matters is how we can represent Chinese judgment document as vectors that can be used in networks.



**Fig. 2.** The architecture of our CNN model

**Model Architecture.** The model architecture, shown in Fig. 2, is a typical simple CNN architecture while using character-level word vectors as its input. Our CNN model has a very small number of layers, which is actually a simplest convolutional neural network. The first layer of our network is the word embedding layer. Each row character vector will be parsed to a low-dimensional vector which has 128 dimensions. The second layer is the convolution layer. A convolution operation can be completed by a lot of filters, which are applied to a window of 5 characters to produce a new feature. In the meanwhile, since our data set is not very large, 256 filters are sufficient.

The third layer is pooling layer. There are two popular pooling methods, mean-pooling and max-pooling, the latter is what we use. The reason why we adopt max-pooling is that we want to find the most favorable features for classification. The fourth and fifth layers are both fully-connected layer. Finally, we predict the probability of the input text belonging to each category through the softmax function.

**Training Process.** There is no doubt that neural networks can not recognize the original text. We think that a very important role of the convolutional neural networks in this classification task is to find favorable features, so we employ one-hot encoding to represent documents instead of TF-IDF. During scanning data sets, we keep 5000 characters which appear most frequently as dictionary. It should be noted that we needn't do any processing on the generated dictionary such as removing pure numbers and so on. We apply convolutional neural networks to this classification task largely because it can learn the favorable feature from text. Therefore, it is considered unnecessary to manually help identify features. After establishing the dictionary, we can convert each character into a one-hot vector which has 5000 dimensions.

The embedding layer of the network is used to complete the word embedding. We use character-level CNN, which needs the information of each character instead of word. In other words, what we need is the features extracted from the text, not the semantic information. So we use the `embedding_lookup` interface from tensorflow to complete the word embedding operation. Specifically, we fully connect 5000 input neurons to 128 neurons in the embedding layer, which completes the work of dimensionality reduction. The weight matrix and bias are trained in the network.

We add dropout module between the first and the second fully-connected layers to regularize. The dropout probability is set to 0.5. The fact proves that the dropout regularization can prevent overfitting very well when the amount of data is not very large.

## 4 Evaluation

In this section, we will answer the three corresponding research questions with experiment results, which can evaluate the contribution made in this paper. Section 4.1 describes the dataset used in Chinese judgment documents classification. Section 4.2 presents results of experiments and answer research questions.

### 4.1 Dataset

As we mentioned earlier, the structure of the Chinese judgment documents in different fields is different. Therefore, we only use judgment documents which are related to liabilities for product quality. We manually divide them into 13 categories according to statutory standard of industry division, including Mechanical Manufacturing, Metal Building Materials, Agricultural, Forestry and Fishing, Chemical, Electronic Communications, Style Supplies, Agricultural and Sideline Food, Textile and Garment, Household Appliance, Others, Pharmaceutical, Transportation and Drinks. It should be noted that only 90 samples belong to Electronic Communications and 131 samples belong to Others.

As for the division of training set, validation set and test set, we use hold-out method. To maximally test the generalization ability of the classifier, we use random sampling method. But considering that the number of samples in some categories may not be sufficient, we draw on the idea of stratified sampling method. If there are too many samples of a certain category be divided into training set while too few samples of it be divided into test set, we will adjust their distribution.

## 4.2 Experiments and Results

**RQ1.** How long is the appropriate length of a sequence of characters extracted from judgment documents? How is the performance of the classifier improved by reducing the sequence length?

Sequence length is a very important factor. Table 1 shows three typical sequence lengths, its time cost and overall accuracy. Obviously 1000 is a suitable sequence length. The overall accuracy reached 87.76%. If you want to train a good classifier in a very short time, then the length of 100 looks good.

**Table 1.** Time cost and overall accuracy between different sequence length

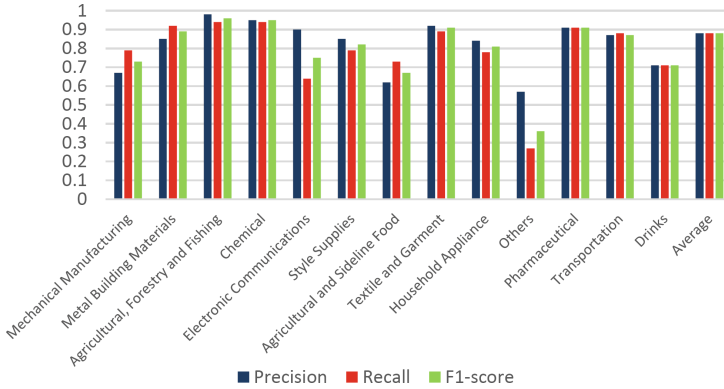
Sequence length	Time cost	Overall accuracy
100	8 min 23 s	82.60%
1000	1 h 46 min 8 s	87.76%
5000	7 h 34 min 33 s	87.63%

**RQ2.** How well can those judgment documents be classified by using character-level convolutional neural networks? What are the specific performances of this method?

The result of experiment shows that our model achieves an overall accuracy of 89.7% at the 10% validation set and 88.2% at the 10% test set. Strictly speaking, 800 pieces of documents seems not enough to prove the performance of our model. So we extract 30% of the shuffled data set to be test set and abandon validation set to further prove the performance of our model. Figure 3 shows the result of experiment which uses 30% of data set as test data. As we can see, the overall accuracy can reach 87.76%.

**RQ3.** Is it a better choice to adopt character-level convolutional neural networks compared with other methods such as word-level convolutional neural networks, recurrent neural networks, Naive Bayes, Decision Tree, Random Forest and Support Vector Machine?

In our previous work [16], a machine learning algorithms-based approach was proposed for Chinese judgment documents classification. Now we use neural networks instead of traditional machine learning algorithms. First of all, from the perspective of training step, the approach we propose this time require less manpower. Since we use character-level CNN, word segmentation, features extraction and feature reduction becomes completely unnecessary. Secondly, the approach we proposed in this paper requires less training time. Finally and most importantly, we can achieve higher overall accuracy, precision, recall and F1-score using character-level CNN. According to the



**Fig. 3.** The specific result of Chinese judgment documents classification

current results, the character-level CNN is better than other machine learning algorithms in this classification task. The details of comparison are shown in Table 2.

**Table 2.** The comparison between character-level CNN and other machine learning algorithms

Algorithms	Time cost	F1-score
NB	2 min 25 s	0.73
DT	2 min 44 s	0.78
RFC	2 min 51 s	0.81
SVM	11 h 44 min 17 s	0.87
Character-level CNN	1 h 46 min 8 s	0.88
Word-level CNN	1 h 28 min 42 s	0.86
Recurrent Neural Networks (GRU)	18 h 20 min 11 s	0.71

## 5 Conclusion and Future Work

The study contributes to the practice of analyzing Chinese judgment documents, specifically speaking, we propose an approach for Chinese judgment documents classification using character-level convolutional neural networks. The characteristics of our work is that we use character-level CNN to classify Chinese judgment documents. It avoids extra errors brought by Chinese word segmentation process. And our model can be easily applied to classification task in other domains. Compared with other machine learning methods such as Support Vector Machine, our model can train better classifier in less time. The average F1-score of our model is 88%.

However, our model still has many limitations. We envisage future research in multiple directions. First, we will study the problem of unbalanced categories. From the experimental results, we can see that for the category with insufficient sample number, sometimes the accuracy and recall are satisfactory, sometimes they are not. Second, in many case, a Chinese judgment document may belong to more than one industry fields.



So multi-label classification task is what we will study. Third, as we all know, neural networks need to do a lot of work to weaken overfitting when the amount of data is not very large. In fact, there are many measures for preventing overfitting, which is what we need to study.

**Acknowledgment.** This work was supported by the National Key R&D Program of China (2016YFC0800803).

## References

1. Carreño, L.V.G., Winbladh, K.: Analysis of user comments: an approach for software requirements evolution. In: ICSE, pp. 582–591 (2015)
2. Hua, W., Wang, Z., Wang, H., Zheng, K., Zhou, X.: Understand short texts by harvesting and analyzing semantic knowledge. *IEEE Trans. Knowl. Data Eng.* **29**(3), 499–512 (2017)
3. Maalej, W., Nabil, H.: Bug report, feature request, or simply praise? On automatically classifying app reviews. In: 23rd IEEE RE, pp. 116–125 (2015)
4. Paul, M.J.: Feature selection as causal inference: experiments with text classification. In: CoNLL, pp. 163–172 (2017)
5. Yang, Y., Yan, Y., Qiu, M., Bao, F.S.: Semantic analysis and helpfulness prediction of text for online product reviews. In: ACL, vol. 2, pp. 38–44 (2015)
6. Li, C., Huang, L., Ge, J., Luo, B., Ng, V.: Automatically classifying user requests in crowdsourcing requirements engineering. *J. Syst. Softw.* **138**, 108–123 (2018)
7. Rousseau, F., Kiagias, E., Vazirgiannis, M.: Text categorization as a graph classification problem. In: ACL, vol. 1, pp. 1702–1712 (2015)
8. Grave, E., Mikolov, T., Joulin, A., Bojanowski, P.: Bag of tricks for efficient text classification. In: EACL, vol. 2, pp. 427–431 (2017)
9. Kim, Y.: Convolutional neural networks for sentence classification. In: EMNL, pp. 1746–1751 (2014)
10. Shang, L., Lu, Z., Li, H.: Neural responding machine for short-text conversation. In: ACL, vol. 1, pp. 1577–1586 (2015)
11. Ahn, S., Choi, H., Pärnamaa, T., Bengio, Y.: A Neural Knowledge Language Model. CoRR abs/1608.00318 (2016)
12. Johnson, R., Zhang, T.: Supervised and semi-supervised text categorization using LSTM for region embeddings. In: ICML, pp. 526–534 (2016)
13. Lai, S., Xu, L., Liu, K., Zhao, J.: Recurrent convolutional neural networks for text classification. In: AAAI, pp. 2267–2273 (2015)
14. Tang, D., Qin, B., Liu, T.: Document modeling with gated recurrent neural network for sentiment classification. In: EMNLP, pp. 1422–1432 (2015)
15. Zhang, X., LeCun, Y.: Text Understanding from Scratch. CoRR abs/1502.01710 (2015)
16. Lei, M., Ge, J., Li, Z., Li, C., Zhou, Y., Zhou, X., Luo, B.: Automatically classify chinese judgment documents utilizing machine learning algorithms. In: DASFAA Workshops, pp. 3–17 (2017)



# RC-CNN: Reverse Connected Convolutional Neural Network for Accurate Player Detection

Lijing Zhang<sup>1,2</sup> , Yao Lu<sup>1,2</sup> , Ge Song<sup>2</sup> , and Hanfeng Zheng<sup>2</sup> 

<sup>1</sup> Beijing Laboratory of Intelligent Information Technology,  
Beijing Institute of Technology, Beijing, China  
focus\_zlj@bit.edu.cn

<sup>2</sup> School of Computer Science, Beijing Institute of Technology, Beijing, China

**Abstract.** Player detection is a valuable but challenging task in computer vision due to the specific application scenes, like motion blur, changing illumination, multi-scale players and so on. To get better detection results, we propose reverse connected modules embedded into the convolutional neural network to pass semantic information captured by deep layers back to shallower layers and integrate features derived from multiple layers into multi-scale features. To better explore the efficiency of our model, we design a group of comparative experiments to discover the rational location to embed our proposed reverse connected module. Through testing and evaluating on the two public dataset: Soccer player and KITTI pedestrian, we verify that our proposed reverse connected convolutional neural network (RC-CNN) can detect multi-scale players in various challenging scenes, which yields an mAP increment of 1.8 points compared to the SSD benchmark.

**Keywords:** Player detection · Reverse connected module  
Multi-scale feature

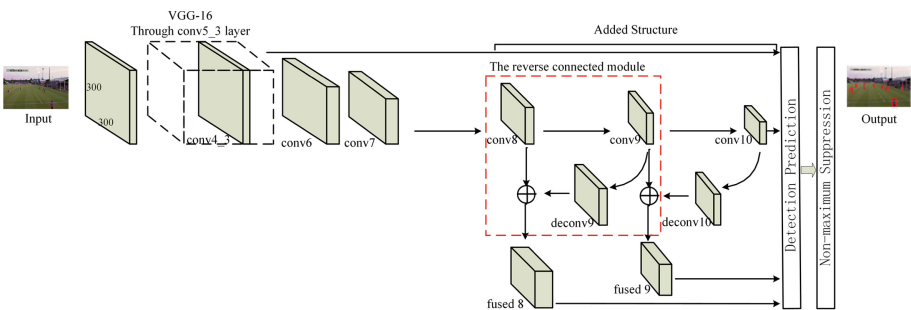
## 1 Introduction

Player detection is the technology that can acquire the spatial locations and relationship of players in every frame. It provides fundamental but crucial information in sports video analysis, and facilitates a number of applications like highlight event detection and team tactical analysis. Due to the specific application scenarios in which the players are playing on the standard playground, the naive method that the background modeling methods [11] adopt to build a static background of the playground with image preprocessing, then to detect the players through background difference method. However, this method is confined to the speed of camera and players' movement. In pursuing the accuracy, some researchers detect players through the methods of learning-based [8, 14] in recent years. However, no matter how accurate the learning-based model is,

motion blur and rapid illumination changes in most shots are Achilles' heel, yielding player detection and tracking less optimal and ambiguous. In addition, players usually are very small and have low resolution, which results in difficulties to get low-level structures and high-level semantic information. As the experiments shown in other papers [1, 7], shallower layers are fit to detect small objects, but the results in small object detection are not very precise because of the lack of semantic information. Therefore, passing the semantic information captured from deeper layers back to the shallower layers will improve the performance of player detection.

Inspired by that, we propose an end-to-end Reverse Connected Convolutional Neural Network (RC-CNN) by using reverse connected modules we designed. And the flow of the model is shown in Fig. 1. Our contributions are summarized as follows:

- Propose reverse connected modules embedded into the convolutional neural network to pass semantic information captured by deep layers back to shallower layers and integrate features derived from multiple layers into multi-scale features.
- Design a group of comparative experiments to explore the rational location to add our proposed reverse connected module. In this way, we verify the efficiency of our proposed model: Reverse Connected Convolutional Neural Network (RC-CNN).
- Conduct the comparative experiments on two datasets: Soccer player [5] and KITTI pedestrian [6]. And the detection results show that our proposed model achieves a superior performance as compared to previous method SSD.



**Fig. 1.** The whole RC-CNN architecture. Inspired by the SSD, our RC-CNN model replaces last two fully-connected layers in VGG-16 with convolutional layers and adds the reverse connected modules to apply the fused feature extracted by multiple layers. And the whole network is optimized by a multi-task loss function.

## 2 Related Work

### 2.1 Player Detection

Player detection is a fundamental task in video analysis and has been addressed by a wide variety of methods. According to the special background, Yu et al. [11] presents a background subtraction algorithm based on Gaussian Mixture Models. More common approaches are based on the feature extractor, Miyamoto et al. [12] utilize color information in an unsupervised manner to improve detection accuracy, but the results are easily contaminated by the motion of camera and the illumination of images. Cascaded classifiers are widely used in detection. For example, Lu et al. [9] trains a binary classification network for labeled image patches and applies the network to achieving a pleasant result, but the training process is a little complicated. Mahmood et al. [10] proposes a system to detect and recognizes the players using the Adaboost algorithm. Through the methods of cascaded classifier, the speed of detection is improved on the trade-off of a little accuracy. All in all, they are not as robust as deep learning based methods.

### 2.2 CNN-Based Object Detection

In recent years, CNN-based object detection shows its superiority obviously in the robust of varying illumination and multi-scale objects. For instance, Girshick et al. [2] makes selective search approach efficiently implemented within a convolutional network. From then on, many region-based methods also compute the location and classification through extracting the CNN feature. In terms of reducing the computation, Ren et al. [14] presents the region proposal network so that the extracted CNN features can be shared in the whole network. Redmon et al. [13] frames object detection as a regression problem and realizes the detection end-to-end through the network which achieves a high detection speed. Inspired by the idea of [13, 14], Liu et al. [8] uses multi-scale CNN features to do the prediction which improves the detection accuracy notably. However, this work is limited in the detection of small objects, because the small object has low resolution and limited semantic information. Kong et al. [7] adds objectness prior networks to predict the object location in advance which is beneficial for reducing the amount of computation.

## 3 Methods

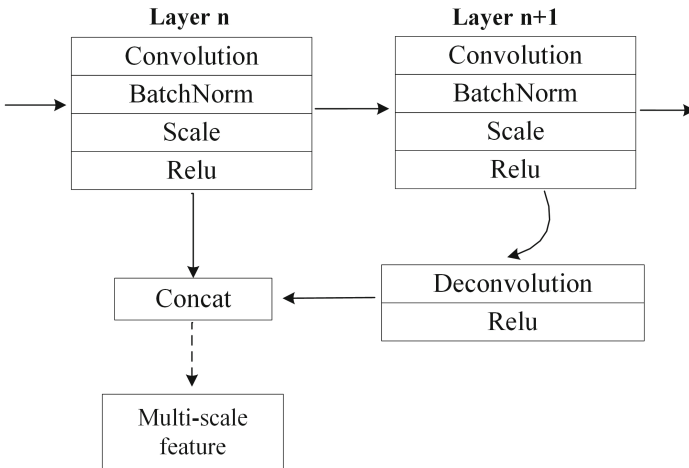
This section describes our proposed Reverse Connected Convolutional Neural Network(RC-CNN). In the first part, the baseline model Single Shot Multibox Detector (SSD) will be briefly introduced. In the second part, the whole network architecture of our model, especially the improvement based on SSD will be illustrated in detail, including the reverse connected module we designed and how to find the proper positions to add it.

### 3.1 The Baseline Model: SSD

The Single Shot Multibox Detector (SSD) [8] is a convolutional neural network used to detect objects, which has been proved to obtain the state-of-the-art. The author uses VGG-16 as the test “base”, which is pre-trained with ImageNet dataset. To satisfy the input images with any sizes, the author replaces the last two fully-connected layers in VGG-16 to convolutional layers. Then SSD adds a series of progressively smaller convolutional layers, meanwhile, applies the feature extracted by multiple layers to predict objects with various scales.

### 3.2 The RC-CNN Model

In line of SSD model, we propose a Reverse Connected Convolutional Neural Network (RC-CNN) to predict the multi-scale players’ location depending on features extracted from multiple convolutional layers. Learned from [15], the features extracted from different layers in the network show the hierarchical nature. That is the shallower layers responds to the information of edge and corner, then more complex invariances, textures and small objects, and then big objects with the layers get deeper. [3, 4] demonstrate that combining fine-grained details with highly-abstracted information helps object detection with different scales. Inspired by the residual connection [4], we develop an architecture of reverse connected module integrated to the convolutional neural network that gets the convolutional computation from deeper layer back to the shallower layer. Through this way, the former layers with lower receptive field could obtain more semantic information and exploit the useful local context in the playing games.



**Fig. 2.** One of the reverse connected modules of our architecture.

One of the reverse connection modules is shown in Fig. 2. Firstly, for the original convolutional layers, in order to improve the generalization ability and

convergence rate of the network, we implement the Batch Normalization. The outputs from each added convolutional layer are all normalized. Then the normalized result will be scaled and shifted after the scale layer. More importantly, a deconvolution layer is applied to passing the forward computation from layer  $n+1$  back, and its output has to be the same scale and dimension with the fused layer  $n$ . Then the two corresponding feature maps are linked through Concat layer. In this way, we get the total feature extracted from the layer  $n+1$  and layer  $n$ , which means the features containing information closer to the semantic information from layer  $n+1$  are passed back to layer  $n$ .

**Where to Add the Reverse Connected Module?** Considering about the scale of players in the shots during the game, we test the specific location to add the reverse connected module to the convolutional network.

The SSD model adds 4 convolutional layers: layer 8 to layer 11 after the modified VGG-16, and achieves a state-of-the-art result. In the model I, the reverse connected module is implemented in between the all 4 convolutional layers. As we all know, the players are relatively small in most shorts. However, the deeper the convolutional layers are, the bigger receptive fields they have. So in the model II, we remove the convolutional layers 11 and add 2 reverse connected module between layer 8 to layer 10. To explore whether more reverse connected module will bring better results, then in the model III, we test adding 3 reversed connected modules between layer 7 to layer 10. The architecture settings of the three models and the SSD model are listed in the Table 1. The “+” represents that we add the reverse connected module between the two layers and “—” denotes that this layer is removed. The model II outperforms others, which is demonstrated through a group of comparative experiments. The implement details about the experiments will be displayed specifically next section. So the model II is referred to as RC-CNN model.

In this case, we minimize the overall objective loss which is a weighted sum of the confidence loss  $L_{conf}$  and the localization loss  $L_{loc}$ .

$$L(x, c, l, g) = \frac{1}{N}(L_{conf}(x, c) + \alpha L_{loc}(x, l, g)) \quad (1)$$

where  $N$  is the number of matched default boxes. A number of images containing multi-scale objects with various challenges are selected to test and evaluate our model.

**Table 1.** The locations of module setting in the comparative experiment

	layer 7	layer 8	layer 9	layer 10	layer 11
SSD300					
model I		+	+		+
model II		+	+		—
model III	+	+	+		—

## 4 Experiments

In this section, we will explain the implement details of experiments including how to conduct the data augmentation and select the negative samples in the training stage. Also, the experiments progress on the two datasets: soccer player [5] and KITTI pedestrian [6] is described specifically. To make the models more robust to various resolution and different object scales, we adopt the method of data augmentation, including color jitter, Gaussian noise and cropped randomly added on each image. During the training phase, the layers in “base” work are initialized by standard VGG-16 model. For the region proposal, all of the positive samples are selected and negative samples are randomly selected to keep the ratio of 1:3.

**Table 2.** The testing results of the comparative experiment.

	Soccer player dataset		KITTI pedestrian dataset	
	mAP(%)	FPS	mAP(%)	FPS
SSD300	73.18	21.92	85.22	35.54
Model I	73.06	21.05	84.20	34.91
RC-CNN model	<b>74.98</b>	<b>22.34</b>	<b>85.79</b>	<b>36.16</b>
Model III	73.85	21.43	84.29	35.97

### 4.1 Soccer Player Dataset

The soccer player dataset is consisted of 2,019 manually annotated images and the height of players in the images is diversified from about 20 pixels to 250 pixels. And all the images are selected from the highlight videos which represent typical soccer games events such as goals, goal attempts, passings and so on.

Table 2 shows our results on the Soccer player dataset. The SSD model achieves 73.38% of the accuracy and 21.92 FPS on this dataset. Compared with the SSD, the mAP of the model I deteriorate slightly. While by removing the layer eleven and utilizing the modules from layer eight to layer ten, the RC-CNN model obtains the result of 74.78% mAP and 22.34 FPS, both of which increase a bit. Then on the base of the RC-CNN model, the model III adds the reverse connected module before layer 8 and achieves an better result than SSD, but less than the RC-CNN model.

As the results show, integrating the reverse connected module into the convolutional neural network is helpful for detecting players. But large receptive fields would often introduce useless background noise, too many modules added may cause the result decreases. And as we collected, the model we proposed is almost the same speed as the original SSD for multiple reasons.

Then we evaluate the models with the performance between the original SSD and our proposed RC-CNN model outperformed mentioned above. Some average results of detection examples on soccer player dataset and other matches

downloaded are shown in Fig. 3. We discuss various of difficult scenarios of player detection during a soccer game: (i) the motion blur caused by the speed of camera and players' movements; (ii) mutual occlusion during the game; (iii) multi-scale players especially the players in the distance; (iv) multi-scale players in a shot. The test results demonstrate that our models can detect more precisely than the original one.



**Fig. 3.** Some detection results on soccer player dataset with various challenges.

## 4.2 KITTI Pedestrian Dataset

To further validate the proposed framework on a larger and more challenging dataset, we conduct experiments on pedestrian dataset collected from KITTI [6]. We cut every image in the dataset along with the width and remove those containing no pedestrians to increase our training sample and reach a total number of 2628 pieces of training samples eventually. In addition, the data augmentation mentioned before is conducted. The experiment procedure is the same as the experiment on the soccer player dataset, and the comparison results of different models are also shown in the Table 2. By comparing the statistical results, We find the our proposed RC-CNN model has an outstanding performance both on detection accuracy and speed.

## 5 Conclusion

This paper proposes a reverse connected convolutional neural network (RC-CNN), an efficient framework for accurate player detection in various challenging



scenes. We design a reverse connected module to make the shallower layers get the semantic information to improve the detection accuracy. Through a group of comparative experiments, we verify the efficiency of the reverse connected module and the rationality of the location added in the convolutional neural network. By testing and evaluating on the two public dataset, our model yields an mAP increment of 1.8 points compared to the SSD benchmark, in addition to reduction of computational cost. Since some players with special posture, like falling to the ground, are not detected, explaining and solving the problem will be our future work.

## References

1. Fu, C.Y., Liu, W., Ranga, A., Tyagi, A., Berg, A.C.: DSSD: deconvolutional single shot detector. In: *Computer Vision and Pattern Recognition* (2017)
2. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: *Computer Vision and Pattern Recognition*, pp. 580–587 (2014)
3. Hariharan, B., Arbelaez, P., Girshick, R., Malik, J.: Hypercolumns for object segmentation and finegrained localization. In: *Computer Vision and Pattern Recognition*, pp. 447–456 (2014)
4. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: *Computer Vision and Pattern Recognition*, pp. 770–778 (2016)
5. Lu, K., Chen, J.J.L., He, H.: [http://www.cs.ubc.ca/~jhchen14/ccnn\\_player\\_detection/](http://www.cs.ubc.ca/~jhchen14/ccnn_player_detection/)
6. KITTI. [http://www.cvlibs.net/datasets/kitti/eval\\_object.php/](http://www.cvlibs.net/datasets/kitti/eval_object.php/)
7. Kong, T., Sun, F., Yao, A., Liu, H., Lu, M., Chen, Y.: RON: reverse connection with objectness prior networks for object detection. In: *Computer Vision and Pattern Recognition* (2017)
8. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: SSD: single shot multibox detector. In: *Computer Vision and Pattern Recognition*, pp. 21–37 (2015)
9. Lu, K., Chen, J., Little, J.J., He, H.: Light cascaded convolutional neural networks for accurate player detection. In: *The British Machine Vision Conference* (2017)
10. Mahmood, Z., Ali, T., Khattak, S.: Automatic player detection and recognition in images using AdaBoost. In: *International Bhurban Conference on Applied Sciences & Technology*, pp. 64–69 (2012)
11. Ming, Y., Guodong, C., Lichao, Q.: Player detection algorithm based on gaussian mixture models background modeling. In: *Second International Conference on Intelligent Networks and Intelligent Systems, ICINIS 2009*, pp. 323–326. IEEE (2009)
12. Miyamoto, R., Oki, T.: Soccer player detection with only color features selected using informed haar-like features. In: Blanc-Talon, J., Distant, C., Philips, W., Popescu, D., Scheunders, P. (eds.) *ACIVS 2016. LNCS, vol. 10016*, pp. 238–249. Springer, Cham (2016). [https://doi.org/10.1007/978-3-319-48680-2\\_22](https://doi.org/10.1007/978-3-319-48680-2_22)
13. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, realtime object detection. In: *Computer Vision and Pattern Recognition*, pp. 779–788 (2015)

14. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards realtime object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **39**(6), 1137–1149 (2017)
15. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) *ECCV 2014*. LNCS, vol. 8689, pp. 818–833. Springer, Cham (2014). [https://doi.org/10.1007/978-3-319-10590-1\\_53](https://doi.org/10.1007/978-3-319-10590-1_53)



# Uncertainty Estimation for Strong-Noise Data

Bin Shen<sup>(✉)</sup> and Binheng Song

Computer Science and Technology, Graduate School at Shenzhen,  
Tsinghua University, Shenzhen 518055, China  
shenbin.ringo@gmail.com, songbinheng@sz.tsinghua.edu.cn

**Abstract.** The measurement of uncertainty in classification tasks is a challenging problem. Bayesian neural networks offer a standard mathematical framework to address the issue but limited by the high computational cost. Recently, several non-Bayesian approaches like *Deep Ensemble* are proposed as alternatives. However, most of the works focus on measuring the model uncertainty rather than the uncertainty over the data. In this paper, we demonstrate that noise in the training data has an adverse impact on uncertainty estimation, and we prove that *Deep Ensemble* is ineffective when training on the strong-noise dataset. We propose an easy-implemented model to estimate the uncertainty on noisy datasets, which is compatible with many existing classification models. We test our method on Fashion MNIST, Fashion MNIST with different levels of Gaussian noise, and the strong-noise financial dataset. The experiments show that our approach is effective on each dataset, whether it contains strong noise or not. The usage of our method improves the trading strategy to increase the annual profit by nearly 5%.

**Keywords:** Uncertainty · Degree of membership  
Classification networks · Strong noise

## 1 Introduction

Most works on classification tasks focus more on improving task performance [1, 2], while the studies on uncertainty have not made a breakthrough. For example, in the medical field, it's significant to know the prediction's confidence to determine how to act upon it [3]. Another typical application is the prediction of financial data, in which each sample contains strong noise. Recent works show that it's difficult to achieve satisfactory accuracy on the whole testing set [4–6] in finance. If the accuracy increases with the confidence scores monotonously, we'll trade more certainly on the predictions with high confidence scores.

Defining and measuring the confidence is the critical problem on this topic. In classification tasks, the absolute value of softmax can be regarded as a simple confidence estimate. However, recent works [7, 8] show that it is ineffective. Bayesian models are used to measure the uncertainty by computing the posterior distribution over the parameters of neural networks [9]. However, the high

computational cost limits its widespread usage. To address the issue, Gal and Ghahramani [10] develop a theoretical framework casting dropout training in deep neural networks as approximate Bayesian inference in deep Gaussian processes. Recently, several non-Bayesian approaches are proposed. Mandelbaum and Weinshall [3] propose a confidence score which is based on the data embedding derived from the penultimate layer of the networks and used a distance-based loss to achieve embedding. Subramanya et al. [7] propose a confidence measurement based on density modeling approaches. Lakshminarayanan et al. [11] offer a scalable approach called *Deep Ensemble* by using ensembles and adversarial training [12] to improve accuracy and obtain uncertainty estimates of the neural networks. *Deep Ensemble* is relatively simple to implement and easy to yield uncertainty estimates leading to its popularity in practice.

As mentioned in Gal’s doctoral thesis [13], uncertainty comes from noise, uncertainty in model parameters, structure uncertainty mainly. The latter two uncertainties can be grouped under model uncertainty. Most works like *Deep Ensemble* focus on model uncertainty without considering the noise from training data, which leads to the ineffectiveness when training models on the noisy dataset. We experimentally verify this phenomenon in Sect. 3.2.

In this paper, we propose a novel model to measure the uncertainty of each prediction, especially on strong-noise datasets. Our contributions are summarized as follows: Firstly, we demonstrate that noise in the training data has a bad impact on the measurement of uncertainty. Secondly, considering that most state-of-the-art models are neural networks, we estimate the uncertainty by embedding a simple neural network into the original model and designing a new objective function, so that the original models can keep their structures.

## 2 Our Method

As mentioned above, uncertainty comes from noise and model uncertainty, but most works are focused on model uncertainty rather than noise in the data. The data with noise contains more uncertainty than the clean one and uncertainty increases with the increase of noise. It is consistent in different kinds of models, in other words, it should be satisfied in different models if not, it means that the models fit too much noise and haven’t learned useful knowledge.

To address the issue above, we use the fuzzy set to define the level of noise in data. For the sample set  $X = \{x_1, x_2, \dots, x_n\}$  where  $x_i \in R^d, i = 1, 2, \dots, n$ . We define a fuzzy set  $F = \{x, \sigma_F(x) | x \in X\}$  in  $X$  as follow.  $F$  is characterized by a membership function  $\sigma_F$  which associates with each point in  $X$  a real number in the interval  $[0, 1]$ , with the values of  $\sigma_F(x)$  at  $x$  representing the “grade of membership” of  $x$  in  $F$ .

$$\sigma_F : X \rightarrow [0, 1] \tag{1}$$

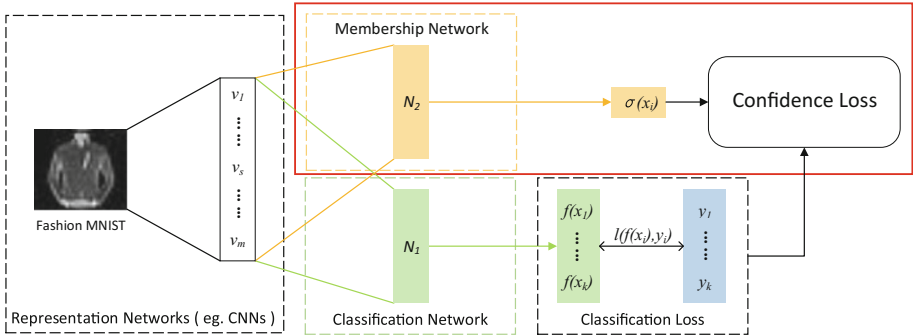
$$\begin{cases} \sigma_F(x) = 1, & x \text{ is clean} \\ \sigma_F(x) = 0, & x \text{ is noise} \\ \sigma_F(x) \in (0, 1), & x \text{ contains noise} \end{cases} \tag{2}$$

$\sigma_F(x)$  defines the level of noise in data, and  $\sigma_F(x)$  increases with the decrease of noise. In our method, we use an embedding neural network to approximate the membership function  $\sigma_F$ .

It should be noted that fuzzy set has been used to estimate uncertainty for a long time, and fuzzy neural networks (FNNs) which combine the fuzzy inference and neural networks are widely used [14]. However, FNNs require large modification on original structures and is hard to reuse existing models. In this paper, we expect to estimate uncertainty in the premise of keeping original structures of classification models so that our method can suit many existing models.

## 2.1 The Architecture

The architecture of our model is summarized in Fig. 1. The structure outside the red box is the original classification neural network, and that inside is what we embed. We first use representation network to extract features, and then, we feed the feature vector to two parallel networks  $N_1$  and  $N_2$ .  $N_1$  is the original fully connected neural network used to classify.  $N_2$  is the regression network we embed to approximate the membership function  $\sigma_F$ . For a supervised classification task, it's easy to calculate the distance between predictions and labels, but there are no labels for the degree of membership. To address this issue, we design a new confidence loss function to learn it.



**Fig. 1.** The architecture of our model.  $l(f(x_i), y_i)$  is the distance between the prediction  $f(x_i)$  and the label  $y_i$  such as the cross entropy loss.  $\sigma(x_i)$  is the degree of membership.

## 2.2 The Loss Function

There are no labels for the degree of membership, so we need to design a novel objective function to estimate it. We discuss two extremes: (1)  $\sigma(x_i)$  approaches 1, it means that  $x_i$  is clean enough so we expect the classification loss  $l(f(x_i), y_i)$  to be minimal; (2)  $\sigma(x_i)$  approaches 0, it means that  $x_i$  is noise and we don't care the classification loss of noise.

We consider multiplying  $\sigma$  with  $l$  as an objective function  $L_m$  to ensure that the classification loss decreases with the decrease of noise.

$$L_m = \sum_{i=1}^n \sigma(x_i) \cdot l(f(x_i), y_i) \quad (3)$$

However,  $L_m$  is not enough,  $\sigma(x_i)$  will approach 0 when training on the data because we haven't controlled the number of clean samples. We expect that there are enough clean samples, so the final loss function is as follow.

$$L = - \sum_{i=1}^n \sigma(x_i) + \lambda \cdot L_m \quad (4)$$

where  $\lambda > 0$ , the left item forces the model to keep enough clean samples, and the right item aims to ensure that low noise leads to low classification loss. The tradeoff between these two terms is captured by a balancing weight  $\lambda$ .

### 2.3 Implement Details of Uncertainty Estimation

$N_1$  and  $N_2$  network are trained at the same time.  $N_2$  is a regression neural network that aims to estimate the membership of each sample. Its output is a single floating number  $\sigma(x_i) \in [0, 1]$ .  $N_1$  is the full connection part of the original model which aims to achieve high accuracy, and we use the negative log likelihood (NLL) as the classification loss. Its output is the loss value which is also a single floating number. And the loss function can be written as follow.

$$L = - \sum_{i=1}^n \sigma(x_i) + \lambda \cdot \sum_{i=1}^n \sigma(x_i) \cdot (-\log p(y_i|x_i)) \quad (5)$$

$\lambda$  is a balancing weight between the number and the classification loss in classifiable samples. If we increase  $\lambda$ , the number of classifiable samples decreases, at the same time, the classification accuracy in this part of samples increases. Unless otherwise specified, we use batch size of 128 and Adam optimizer with the fixed learning rate of 0.001. We train models on one GeForce GTX TITAN X GPU and use the default weight initialization in Keras in our experiments.

Noise is a common source of uncertainty and is consistent in different kinds of models. We estimate the degree of membership to know the level of noise in each sample. As mentioned above, uncertainty increases with the increase of noise, so we reasonably assume that there is a monotonic function that can map the degree of membership to the uncertainty. In this paper, we use the grade of the membership as the uncertainty directly.

## 3 Experiments and Results

### 3.1 Fashion MNIST

Fashion MNIST is a dataset of Zalando's article images [15]—consisting of a training set of 60,000 examples and a test set of 10,000 examples. Each example

is a 28\*28 grayscale image, associated with a label from 10 classes. We split 6,000 examples from the training set as a validation set randomly.

In this experiment, we train original, deep ensemble and our models on Fashion MNIST to verify if the models can estimate the available confidence on the low-noise dataset. We train these models for 40 epochs, and we stop training when the validation loss is no longer decreasing. Table 1 summarizes the results and it shows that all the models can learn the effective confidence. The classification accuracy increases with the threshold of confidence (the decrease of the uncertainty) monotonously.

**Table 1.** Comparison of original, ensemble and our models on Fashion MNIST. Threshold is the threshold of the confidence. The values in the table show the accuracy in different threshold. Keys: c, convolutional layer; p, pooling layer; d, dropout layer; fc, fully connected layer. All the kernel size is (3, 3) and the pool size is (3, 3).

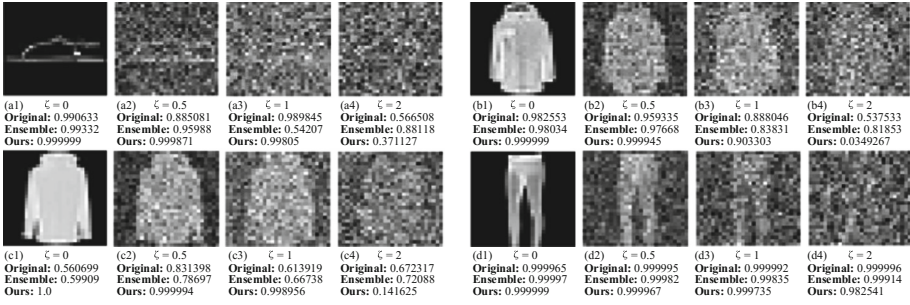
Methods	Architecture	Threshold				
		0.2	0.4	0.6	0.8	1.0
MLP	512fc-d-512fc-d	93.63%	98.05%	99.50%	99.95%	100%
CNN	32c-32c-p-d-64c-64c-p-d-256fc-d-256fc	96.70%	99.15%	99.88%	100%	100%
Ensemble_MLP	MLP * 5	94.61%	98.30%	99.50%	100%	100%
Ensemble_CNN	CNN * 5	97.33%	99.40%	99.98%	100%	100%
Our MLP	MLP + 64fc-32fc	92.79%	94.32%	95.38%	95.90%	97.00%
Our CNN	CNN + 32fc-16fc	93.43%	95.90%	97.65%	98.90%	99.60%

### 3.2 Fashion MNIST with Gaussian Noise

We add different levels of Gaussian noise ( $\zeta \times N(0, 1)$ ) into Fashion MNIST to verify if the models are effective on the noisy dataset, where  $\zeta$  is the noise factor to control the noise levels. There are 54,000 training samples, 6,000 validation samples, and 10,000 testing samples in Fashion Mnist. We add three different levels of Gaussian noise into the dataset and concatenate all samples so that we get a training set of 216,000 samples, a validation set of 24,000 samples and a testing set of 40,000 samples. We train a CNN model of 32c-p-64c-p-10fc (see the ‘Keys’ in Table 1), an ensemble CNN model and our model on the new dataset.

We choose several images from the testing set randomly and feed them to these three trained models. The results are shown in Fig. 2.

The result shows that the original model and the deep ensemble model are confused on the image a, c and d while our model still works. This experiment shows that noise in the training data has a bad impact on the measurement of uncertainty, and it’s not enough just to estimate the model uncertainty, we need to model the uncertainty from noise.



**Fig. 2.**  $a \sim d$  are four images chosen randomly,  $\zeta$  is the level of Gaussian noise, the values in the figure are the confidence scores estimated by each model.

### 3.3 Daily Data of NYSE Stocks

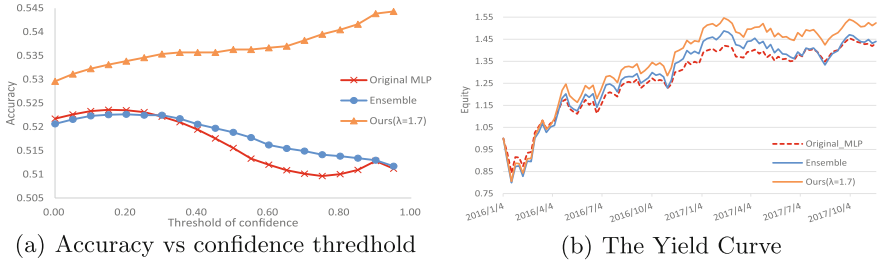
**Dataset Preparation.** We build a stock data set by selecting 2617 NYSE stocks' daily data. We first download all historical data between Jan. 01, 2010 and Dec. 01, 2017 from Yahoo Finance, which is recognized as a reliable source of stock data. We then clean the data and divide it into the training part and the testing part on 2016.01.01. We label the training samples whose weekly return  $> 0.005$  as 1 while it  $< -0.005$  as 0 to filter out the inapparent samples. And we label the testing samples whose weekly return  $> 0$  as 1, otherwise as 0.

There are 3,389,239 samples in the training set and 1,126,055 samples in the testing set. Each sample contains seven dimensions: symbol code, date, opening price, high price, low price, closing price and trading volume. We calculate several technical indicators to extract features of the historical stock data. Indicators in finance are like features in computer vision. To better represent the features of the historical data, we select different types of technical indicators such as momentum indicators, volatility indicators, and others [16]. Indicators contain the information before the given day, so we use the data with indicators as the input rather than time lags. We prepare the data by subtracting the mean and dividing the standard deviation on each stock.

**Experimental Result.** In this experiment, it's inappropriate to use CNNs to extract features because the stock data is sequential without the strong spatial relationship. We calculate several indicators to extract features of the historical data and assume that indicators contain the transaction information before the given day, so it's justified to use the MLP model without representation networks in this application. In this experiment, we use the batch size of 2048 and train each model for 10 epochs. The results are summarized in Fig. 3(a).

We build a trading strategy which selects 300 stocks and readjusts the portfolio in 5 days cycle. We sort the outputs by the confidence score and select top 300 stocks that are classified as 1 to trade. We keep the position of the selected stocks which are already in the portfolio. If the stocks we selected are not in the portfolio, we buy them with the same position. And we sell the stocks which





**Fig. 3.** (a) shows the accuracy under different confidence threshold on the NYSE stock dataset. (b) shows the yield curve of these models in recent two years. It should be noted that we don't remove the transaction cost in this experiment.

are not in the stocks we picked. As Table 2 shows, we use the annual return, the max drawdown, and Sharpe ratio to compare the performances of these three models. The annual return is a return over a period of one year. The max drawdown is the maximum of the drawdown which is the measure of the decline from a historical peak. Sharpe ratio is the average return earned in excess of the risk-free rate per unit of volatility or total risk. To show the comparison clearly, we summarize the accumulative profits of these three methods in Fig. 3(b).

**Table 2.** Comparison of the original model, the ensemble model and ours

Methods	Annual return	Max drawdown	Sharpe ratio
Original MLP	22.55%	<b>15.63%</b>	0.9743
Deep Ensemble	22.95%	20.10%	0.8223
Ours	<b>27.34%</b>	19.47%	<b>1.0144</b>

In this experiment, we aim to select several stocks which are predicted to rise with high confidence, rather than to achieve the state-of-the-art on the whole dataset. Table 2 shows that the original model achieves the annual return of 22.55% in the backtest range from 2016.01.01 to 2017.12.01, while our model achieves 27.34%. The annual profit of our method increases by 4.79% compared with the original model and increases by 4.39% compared with the ensemble model. Although the maximum drawdown of our model is larger, the Sharpe ratio is also greater. It means that our model can seek profits more stable.

## 4 Conclusion and Future Work

In this paper, we demonstrate that noise in training data has an adverse impact on uncertainty estimation. We propose a novel method to estimate uncertainty on strong-noise datasets. In our method, we embed a regression neural network

to approximate the membership function and design a new loss function to learn it. The method is compatible with many existing neural network models, and original models can keep their structures, even keep the original loss function.

Experimental results show that our model can estimate the uncertainty on strong-noise dataset reliably, and the grade of membership about noise can be an effective measurement of uncertainty. And in financial data, the usage of our method increases the annual profit by nearly 5%.

In the future, we expect to model the problem on both noise and model uncertainty so that the uncertainty can be estimated more exactly. We use the degree of membership as the uncertainty directly in this paper, and we'll further discuss the function mapping the membership to uncertainty. Moreover, we expect to apply our method to medical datasets such as MRI (Magnetic Resonance Imaging) datasets to assisting doctors in medical diagnosis.

## References

1. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: NIPS, vol. 60, pp. 1097–1105 (2012)
2. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR, pp. 770–778 (2016)
3. Mandelbaum, A., Weinshall, D.: Distance-based confidence score for neural network classifiers. arXiv preprint [arXiv:1709.09844](https://arxiv.org/abs/1709.09844) (2017)
4. Qiu, M., Song, Y.: Predicting the direction of stock market index movement using an optimized artificial neural network model. PloS One **11**(5), e0155133 (2016)
5. Kara, Y., Boyacioglu, M.A., Baykan, Ö.K.: Predicting direction of stock price index movement using artificial neural networks and support vector machines. Expert Syst. Appl. **38**(5), 5311–5319 (2011)
6. Devadoss, A.V., Ligorì, T.A.A.: Stock prediction using artificial neural networks. Int. J. Data Min. Tech. Appl. **2**, 283–291 (2013)
7. Subramanya, A., Srinivas, S., Babu, R.V.: Confidence estimation in deep neural networks via density modelling. arXiv preprint [arXiv:1707.07013](https://arxiv.org/abs/1707.07013) (2017)
8. Bendale, A., Boulton, T.E.: Towards open set deep networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1563–1572 (2016)
9. Neal, R.M.: Bayesian Learning for Neural Networks, vol. 118. Springer Science & Business Media (2012)
10. Gal, Y., Ghahramani, Z.: Dropout as a Bayesian approximation: representing model uncertainty in deep learning. In: ICML, pp. 1050–1059 (2016)
11. Lakshminarayanan, B., Pritzel, A., Blundell, C.: Simple and scalable predictive uncertainty estimation using deep ensembles. In: NIPS, pp. 6405–6416 (2017)
12. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. arXiv preprint [arXiv:1412.6572](https://arxiv.org/abs/1412.6572) (2014)
13. Gal, Y.: Uncertainty in Deep Learning. University of Cambridge (2016)
14. Chen, C.P., Liu, Y.J., Wen, G.X.: Fuzzy neural network-based adaptive control for a class of uncertain nonlinear stochastic systems. IEEE Trans. Cybern. **44**(5), 583–593 (2014)
15. Xiao, H., Rasul, K., Vollgraf, R.: Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms. arXiv preprint [arXiv:1708.07747](https://arxiv.org/abs/1708.07747) (2017)
16. Pring Martin, J.: Technical Analysis, Explained ‘The Successful Investors’ Guide to Spotting Investment Trends and Turning Points. McGraw-Hill, New York (1991)



# Reciprocal Ranking: A Hybrid Ranking Algorithm for Reciprocal Recommendation

Yuanhang Qu, Hongzhi Liu<sup>(✉)</sup>, Yingpeng Du, and Zhonghai Wu

School of Software and Microelectronics,  
Peking University, Beijing 102600, People's Republic of China  
{qu\_yh, liuhz, dyp1993, wuzh}@pku.edu.cn

**Abstract.** Reciprocal recommendation is an important class of recommendation. It is the core of many social websites like online dating, online recruitment and so on. Different from item-to-people recommenders which only need to satisfy the preference of users, reciprocal recommenders match people and people while trying to satisfy the preferences of both parties. For each user, we provide a ranking list while trying to increase the click rate as well as the probability that clicks receiving positive replies (reciprocal interactions). Most existing methods only consider either unilateral clicks or reciprocal interactions to make recommendation. Few methods consider both of these kinds of information. In this paper, we propose a novel reciprocal recommendation method called Reciprocal-Ranking (RRK), which combines the prediction of unilateral clicks and reciprocal interactions. Experimental results on both a real-world dataset and a synthetic dataset show that RRK performs better than several state-of-the-art methods.

**Keywords:** Reciprocal recommender · Online dating · Learning to rank

## 1 Introduction

Recommenders have been widely used in different domains beyond the traditional item-to-people domains. A new recommendation problem, called reciprocal recommendation or people-to-people recommendation, has emerged in large online markets. For convenience, we call the item-to-people recommendation as unilateral recommendation. Pizzato [1] had firstly given some main difference between reciprocal recommender and unilateral recommender. In unilateral recommendations, success is determined solely by one user. But in reciprocal recommendations, success is determined by both users. In most unilateral recommendations, satisfied users are likely to repeatedly use the site. In contrast, in some reciprocal domains users may leave the site permanently after a successful recommendation. For example, in recruitment websites, employees who have found a long-term job will leave the website.

Most related works provided recommendations according to users' historical behaviors. In the reciprocal problem, user behaviors can be divided into unilateral feedback and reciprocal feedback. Unilateral feedback include clicking, inviting and giving a high score to another user. Reciprocal feedback can be observed as an invitation with a positive reply, or a pair of high scores. According to the information used

by recommender, we could divide recommenders into unilateral feedback based recommender (UFBR), reciprocal feedback based recommender (RFBR), and hybrid feedback based recommender (HFBR).

UFBRs [1–4] always learn users’ unilateral preference from users’ historical unilateral feedback independently, then calculate if two users can match based on the unilateral predictions in simple and naïve ways. In most UFBRs, a correct prediction of reciprocal feedback requires two correct predictions of unilateral feedback. The precision of combinatorial system could drop rapidly. Moreover, some patterns of reciprocal interactions may have to be learned from historical reciprocal interactions. Modeling users independently always loses the important information.

RFBRs focus on the prediction of reciprocal feedback while ignoring users’ unilateral feedback [6]. As said before, some two side markets can’t generate enough reciprocal interactions for recommenders to learn because users may leave after a success interaction. Obviously, enough historical reciprocal feedback could help RFBRs working efficiently, but the data is usually too sparse to train models. Some scholars use clustering to handle the problem of sparseness [5], but clustering could depress the personalization of recommender. HFBRs [7–9, 15] comprehensive consider unilateral feedback and reciprocal feedback. Properly combining different feedback could handle front problems. But exiting HFBRs always have other problems like low personalization, low coverage rate and so on.

We present a general optimizing algorithm called Reciprocal Ranking (RRK) to solve above problems. Our works could be summarized as:

1. We propose RRK for reciprocal recommendation. RRK derived from the maximum posterior probability estimation to maximizing probability of reciprocal feedback and unilateral feedback.
2. RRK belongs to HFBR, which could solve the problems of UFBRs and RFBRs. Moreover, RRK avoids the insufficiency of existing HFBRs.
3. We discuss the evaluation of reciprocal recommendation: with the premise of meeting the prediction of bilateral interaction, user’s unilateral preference should be satisfied as much as possible. In evaluations, not only should the reciprocal interaction be used, but unilateral feedback should also be consulted.
4. We did experiments on real world dataset and synthetic dataset. The RRK can increase performance significantly in prediction of reciprocal feedback as well as in predictions of unilateral feedback.

## 2 Problem Definition

Let  $U$  and  $V$  be two group of users. Each user  $u$  in  $U$  and  $v$  in  $V$  has personalized preference. We use  $u \rightarrow v$  to denote a unilateral feedback, it can be observed as  $u$  invites  $v$  or  $u$  replies  $v$ ’s invitation. Accordingly, we define the unilateral feedback  $p$  below:

$$p_{uv} = \begin{cases} 1, & \text{if } u \rightarrow v \text{ is observed} \\ 0, & \text{else} \end{cases} \quad (1)$$

If  $u$  likes  $v$  and  $v$  likes  $u$ , we call it a match or a reciprocal feedback. Let  $m$  denote the status of match between  $U$  and  $V$ .

$$m_{uv} = \begin{cases} 1, & p_{uv} = 1 \wedge p_{vu} = 1 \\ 0, & \text{else} \end{cases} \quad (2)$$

When  $p_{uv} = 0$ , it means  $u \rightarrow v$  can't be observed in historical data. The problem is recommending each  $u \in U$  a list  $Q_u \subset \{v \in V | p_{uv} = 0 \wedge p_{vu} = 0\}$ , which is sorted by the probability of  $m_{uv} = 1$  and the probability of  $p_{uv} = 1$  in the future.

In the recommending list  $Q_u$ , if  $v_m$  has high probability to match with  $u$ ,  $v_m$  should be placed in the head. Meanwhile  $v_p$  which has high probability to be unilateral liked by  $u$  should be placed before  $v_n$  that  $u$  don't like. Although  $v_p$  may not like  $u$ , these candidates in recommending list could increase users' clicks. Candidates in an ideal recommending list not only has a high probability of matching with  $u$ , but could also improve the unilateral satisfaction of  $u$  as much as possible.

### 3 Reciprocal Ranking

#### 3.1 Assumptions

Let  $q_{uv}$  denote the probability that  $u$  and  $v$  prefer to each other, which called a match or a reciprocal feedback. We can understand  $q_{uv}$  as matching degree, obviously  $q_{uv}$  is related to  $p_{uv}$  and  $p_{vu}$ .

$$q_{uv} := f(p_{uv}, p_{vu}) \quad (3)$$

For a pair of users  $u$  and  $v$ , if  $u$  likes another, the probability of they like each other always increases. To say the least, the probability of match can't decrease when  $u$ 's unilateral preference increase. Accordingly, we can give our base assumption.

*Assumption.* The probability of match is nondecreasing on unilateral preference  $p_{uv}$  and  $p_{vu}$ .

$$\frac{\partial q_{uv}}{\partial p_{uv}} \geq 0, \quad \frac{\partial q_{uv}}{\partial p_{vu}} \geq 0 \quad (4)$$

For user  $u$ , the user set  $V$  can be divided into three subsets, in which  $M_u$  denotes match set,  $P_u$  denotes unilateral positive set and  $N_u$  denotes negative set:

$$\begin{cases} M_u = \{v | p_{uv} > 0 \ \& \ p_{vu} > 0\} \\ P_u = \{v | p_{uv} > 0 \ \& \ p_{vu} = 0\} \\ N_u = \{v | p_{uv} = 0 \ \& \ p_{vu} = 0\} \end{cases} \quad (5)$$

According to definition of  $q_{uv}$ , we can get such order relationship while  $v_m \in M_u$ ,  $v_p \in P_u$  and  $v_n \in N_u$ :

$$q_{uv_m} \geq q_{uv_p} \geq q_{uv_n} \tag{6}$$

### 3.2 Objective Function

Because of the multiple classification, we try to optimize a generalized AUC [10, 11]:

$$GAUC = \frac{1}{\sum_{c_k < c_l} n_k n_l} \sum_{y_i < y_j} \delta(f(x_i) < f(x_j)) \tag{7}$$

In which  $c_k$  and  $y_i$  denote ordinal class labels,  $n_k$  is the number of examples in class  $c_k$ ,  $f(x_i)$  denotes the prediction value of  $x_i$ ,  $\delta(\cdot)$  is indicative function as below:

$$\delta(x > 0) := \begin{cases} 1, & x > 0 \\ 0, & otherwise \end{cases} \tag{8}$$

Because such function is difficult to optimizing, we adopt a widely used function  $\ln\sigma(\cdot)$  to replace the indicative function. Then the GAUC of this specific problem become smooth:

$$GAUC = \frac{1}{N} \sum_{u \in U} \left[ \sum_{v_m \in M_u} \sum_{v_n \in N_u} \ln \sigma(q_{uv_m} - q_{uv_n}) + \sum_{v_m \in M_u} \sum_{v_p \in P_u} \ln \sigma(q_{uv_m} - q_{uv_p}) + \sum_{v_p \in P_u} \sum_{v_n \in N_u} \ln \sigma(q_{uv_p} - q_{uv_n}) \right] \tag{9}$$

Where  $N = \sum_{u \in U} (|M_u| \cdot |P_u| + |P_u| \cdot |N_u| + |M_u| \cdot |N_u|)$ .

To improve the optimization iterative epoch, we can adjust the order relationship of GAUC. We use  $q_{uv_m}, q_{uv_p} > q_{uv_n}$  and  $q_{uv_m} > q_{uv_p}, q_{uv_m} > q_{uv_n}$  to replace  $q_{uv_m} > q_{uv_n}$ ,  $q_{uv_p} > q_{uv_n}$  and  $q_{uv_m} > q_{uv_p}$ . Accordingly, we can give the objective function of reciprocal ranking, which defined as  $\max_{\Theta} RRK - OPT$ , where  $\Theta$  denotes parameters of model:

$$RRK - OPT = \sum_{u \in U} \left[ \frac{\sum_{v_i \in M_u} \sum_{v_j \in P_u \cup N_u} \ln \sigma(q_{uv_i} - q_{uv_j}) + \sum_{v_k \in M_u \cup P_u} \sum_{v_l \in N_u} \ln \sigma(q_{uv_k} - q_{uv_l})}{\sum_{v_i \in M_u} \sum_{v_j \in P_u \cup N_u} \ln \sigma(q_{uv_i} - q_{uv_j}) + \sum_{v_k \in M_u \cup P_u} \sum_{v_l \in N_u} \ln \sigma(q_{uv_k} - q_{uv_l})} \right] - \lambda \|\Theta\|^2 \tag{10}$$

### 3.3 Model Learning

Given the objective function, we use the stochastic gradient descent method to learn the parameters, each group of samples contains an user of set  $U$  and three users of set  $V$ , where  $v_i \in M_u, v_j \in P_u \cup N_u, v_k \in M_u \cup P_u, v_l \in N_u$ .

$$samples = (u, v_i, v_j, v_k, v_l) \quad (11)$$

The gradient of parameters related to each sample is given below:

$$\begin{aligned} \frac{\partial RRK - OPT}{\partial \theta} &= \frac{\partial}{\partial \theta} \ln \sigma(q_{uv_i} - q_{uv_j}) + \frac{\partial}{\partial \theta} \ln \sigma(q_{uv_k} - q_{uv_l}) - \frac{\partial \lambda \|\theta\|^2}{\partial \theta} \\ &= \frac{1}{1 + e^{q_{uv_i} - q_{uv_j}}} \times \frac{\partial (q_{uv_i} - q_{uv_j})}{\partial \theta} + \frac{1}{1 + e^{q_{uv_k} - q_{uv_l}}} \times \frac{\partial (q_{uv_k} - q_{uv_l})}{\partial \theta} - 2\lambda\theta \end{aligned} \quad (12)$$

While the optimizing algorithm can apply to numeric model, we give concrete learning algorithm with matrix factorization (MF). The MF is employed to estimate the probability of match between two users, which is denoted by  $q_{uv}$ . Parameters related to  $q_{uv}$  include the latent feature vector  $W_u$  of the user  $u$  and the latent feature vector  $H_v$  of the user  $v$ .

$$q_{uv} = W_u H_v^T = \sum_f w_{uf} h_{vf} \quad (13)$$

According to Eq. 13 we give the concrete gradient of  $RRK$  on each parameter related to Eq. 12, where  $v_i \in M_u$ ,  $v_j \in P_u \cup N_u$ ,  $v_k \in M_u \cup P_u$ ,  $v_l \in N_u$ .

$$\frac{\partial RRK}{\partial w_{uf}} = \frac{h_{vif} - h_{vjf}}{1 + e^{q_{uv_i} - q_{uv_j}}} - \frac{h_{vkf} - h_{vlf}}{1 - e^{q_{uv_k} - q_{uv_l}}} - 2\lambda w_{uf} \quad (14)$$

$$\frac{\partial RRK}{\partial h_{vif}} = \frac{w_{uf}}{1 + e^{q_{uv_i} - q_{uv_j}}} - 2\lambda h_{vif} \quad (15)$$

$$\frac{\partial RRK}{\partial h_{vjf}} = \frac{-w_{uf}}{1 + e^{q_{uv_i} - q_{uv_j}}} - 2\lambda h_{vjf} \quad (16)$$

$$\frac{\partial RRK}{\partial h_{vkf}} = \frac{w_{uf}}{1 - e^{q_{uv_k} - q_{uv_l}}} - 2\lambda h_{vkf} \quad (17)$$

$$\frac{\partial RRK}{\partial h_{vlf}} = \frac{-w_{uf}}{1 - e^{q_{uv_k} - q_{uv_l}}} - 2\lambda h_{vlf} \quad (18)$$

The pseudocode of  $RRK$  is shown in Algorithm 1. For each user  $u$ , we divide user set  $V$  into three parts  $M_u$ ,  $P_u$  and  $N_u$  according to historical feedback. The parameters  $W$  and  $H$  are set randomly. Then we randomly pick a sample  $(u, v_i, v_j, v_k, v_l)$  where  $v_i, v_j, v_k, v_l$  are picked from  $M_u, P_u$  and  $N_u$  according to  $u$ . Then we use  $RRK - OPT$  to update  $W$  and  $H$  with the gradients given in Eqs. (14, 15, 16, 17 and 18).

---

**Algorithm 1:** RRK Optimizing

---

**Input:** user set:  $U$  and  $V$ , historical feedback:  $u \rightarrow v$  or  $v \rightarrow u$ , number of hidden dimensions:  $F$ , Learning Rate:  $\eta$ , times of sampling:  $T$ .

**Output:** parameters  $\Theta$

Randomly initialize parameters  $W$  and  $H$ .

for each  $u \in U$  do

    Initialize match set  $M_u$ , unilateral positive set  $P_u$  and negative set  $N_u$

end for

for  $s = 0$  to  $T$  do:

    Randomly pick users  $u \in U, v_i \in M_u, v_j \in P_u \cup N_u, v_k \in M_u \cup P_u, v_l \in N_u$

    for each  $\theta \in \Theta$  which is related to samples:

$$\theta = \theta + \eta \cdot \frac{\partial RRK-OPT}{\partial \theta}$$

    end for

end for

---

## 4 Experiment

### 4.1 Data

In order to validate our algorithm, we conducted several experiments with real world dataset and synthetic dataset. We divided each dataset into a training set, a validation set, and a test set. The training set was used for optimizing parameters, the validation set was used for tuning the super parameter, and the test set was used for final evaluation.

**Real World Data.** We used a public dataset to test algorithms, the data was provided by a Czech online-dating website Libimseti. It contains more than  $1.5 \times 10^6$  ratings with values from 1 to 10. We used male users as  $U$  in the assumptions and used female users as  $V$ . Scholars always assumed that rates greater than 5 are similar to positive feedback and we went native [13, 14]. For the convenience of verification, we held users who had at least 3 reciprocal feedback.

**Synthetic Data.** We generated synthetic data based on the empirical assumptions of real world. We supposed the level of user  $u$ 's preference to user  $v$  obeying function:  $p_{uv} = \sum_f b_{uf} a_{vf} + \sigma$ . For example in online dating,  $a_{vf}$  can denote if user  $v$  is a male while  $b_{uf}$  denoting if user  $u$  likes man. The relation between unilateral preference  $p_{uw}, p_{vu}$  and probability of match  $q_{uv}$  was unknown, but for example we assumed  $q_{uv} = p_{uw} + p_{vu} + \sigma$  (Table 1).



**Table 1.** Number of users and feedback

Dataset	User set	Number	Unilateral feedback	Reciprocal feedback
Real data	Male	3813	132705	51942
	Female	3784	215361	
Synthetic data	User set $U$	2166	258948	65062
	User set $V$	2164	260830	

### 4.2 Evaluation Metrics

We adopted some widely used metrics to study the empirical performance of RRK.  $Prec@k$ ,  $F1@k$ , NDCG, MAP and AUC. As we discussed before, we studied these metrics on predictions of both reciprocal feedback and unilateral feedback.

### 4.3 Result and Analysis

We compared the algorithm with CSVD [9], IBCF [6], and BPR-MF [12]. CSVD and IBCF were described before, which were used for reciprocal recommendation by other researchers. Because our algorithm used Learning to Rank method to optimize parameters, we think it’s important to compared with a well-known Learning to Rank algorithm like BPR.

**Result on Predictions of Reciprocal Feedback.** According to the experiments, RRK produced better results on real data and synthetic data. Compared with the best performance of baseline algorithms, the performance of RRK are over 14%, 17%, 2%, 5%, 10% for  $Prec@5$ ,  $F1@5$ , AUC, NDCG and MAP (Table 2).

**Table 2.** Reciprocal performance improvement of RRK over baseline algorithms

	Real data					Synthetic data				
	Prec@10	F1@10	AUC	NDCG	MAP	Prec@10	F1@10	AUC	NDCG	MAP
BPR	0.042	0.056	0.768	0.248	0.039	0.101	0.093	0.871	0.342	0.066
IBCF	0.041	0.054	0.637	0.243	0.045	0.121	0.116	0.636	0.355	0.081
CSVD	0.005	0.006	0.507	0.171	0.006	0.055	0.040	0.593	0.264	0.028
RRK	0.048	0.066	0.826	0.271	0.053	0.145	0.137	0.889	0.376	0.090
Over best	14.32%	17.94%	7.55%	9.35%	18.44%	19.38%	18.78%	2.08%	5.88%	10.51%

**Result on Predictions of Unilateral Feedback.** As we discussed in the problem setting, the candidates in an ideal recommending list not only have a high probability of matching with  $u$ , but could also improve the unilateral satisfaction of  $u$  as much as possible. Therefore, we had evaluated the prediction of unilateral feedback. Such targets aren’t the most important targets, but they can help to increase users’ unilateral satisfaction. While coming to the unilateral goal, the algorithm gave more obvious advantages (Table 3).

**Table 3.** Unilateral performance improvement of RRK over baseline algorithms

	Real data					Synthetic data				
	Prec@10	F1@10	AUC	NDCG	MAP	Prec@10	F1@10	AUC	NDCG	MAP
BPR	0.100	0.079	0.740	0.349	0.046	0.257	0.074	0.776	0.498	0.088
IBCF	0.088	0.070	0.616	0.334	0.044	0.283	0.088	0.612	0.488	0.072
CSVD	0.010	0.006	0.504	0.258	0.009	0.134	0.030	0.555	0.428	0.048
RRK	0.189	0.146	0.865	0.430	0.102	0.389	0.125	0.849	0.554	0.135
Over best	89.86%	84.34%	16.86%	23.34%	119.62%	37.79%	42.04%	9.31%	11.10%	53.68%

## 5 Conclusion

In this paper, we have proposed a reciprocal ranking algorithm (RRK). It's an optimization algorithm applicable for reciprocal recommender. The RRK combines unilateral feedback and reciprocal feedback to predict probability of reciprocal feedback. According to theoretical derivation and experiment, the RRK is better than UFBR, RFBR and existing HFBR. In this paper, we only use information about users' behavior because of the limit of dataset. To solve the cold start problem, we are interested in extending RRK to take consideration of other auxiliary information such as the explicit preference and personal profile of users in future works.

**Acknowledgments.** This work was partially sponsored by National Key R&D Program of China (Grant No. 2017YFB10 020 02) and PKU-Tencent joint research Lab.

## References

1. Pizzato, L., Rej, T., et al.: RECON: A reciprocal recommender for online dating. In: Proceedings of the fourth ACM conference on Recommender systems, pp. 207–214. ACM, New York (2010)
2. Kim, Y.S., et al.: People recommendation based on aggregated bidirectional intentions in social network site. In: Kang, B.-H., Richards, D. (eds.) PKAW 2010. LNCS (LNAI), vol. 6232, pp. 247–260. Springer, Heidelberg (2010). [https://doi.org/10.1007/978-3-642-15037-1\\_21](https://doi.org/10.1007/978-3-642-15037-1_21)
3. Kutty, S., Chen, L., et al.: A people-to-people recommendation system using tensor space models. In: Proceedings of the 27th Annual ACM Symposium on Applied Computing, pp. 187–192. ACM, New York (2012)
4. Tu, K., Ribeiro, B., et al.: Online dating recommendations: Matching markets and learning preferences. In: International Conference on World Wide Web, pp. 787–792. ACM, New York (2014)
5. Alsaleh, S., Nayak, R., Xu, Y., Chen, L.: Improving matching process in social network using implicit and explicit user information. In: Du, X., Fan, W., Wang, J., Peng, Z., Sharaf, M.A. (eds.) APWeb 2011. LNCS, vol. 6612, pp. 313–320. Springer, Heidelberg (2011). [https://doi.org/10.1007/978-3-642-20291-9\\_32](https://doi.org/10.1007/978-3-642-20291-9_32)
6. Krzywicki, A., et al.: Interaction-Based collaborative filtering methods for recommendation in online dating. In: Chen, L., Triantafillou, P., Suel, T. (eds.) WISE 2010. LNCS, vol. 6488, pp. 342–356. Springer, Heidelberg (2010). [https://doi.org/10.1007/978-3-642-17616-6\\_31](https://doi.org/10.1007/978-3-642-17616-6_31)

7. Akehurst, J., Koprinska, I., et al.: CCR: A content-collaborative reciprocal recommender for online dating. In: Toby W. (eds.) *International Joint Conference on Artificial Intelligence*, vol. 3, pp. 2199–2204. AAAI Press (2011)
8. Kutty, S., Nayak, R., et al.: A people-to-people matching system using graph mining techniques. *World Wide Web* **17**(3), 311–349 (2014)
9. Ting, C.-H., Lo, H.-Y., Lin, S.-D.: Transfer-learning based model for reciprocal recommendation. In: Bailey, J., Khan, L., Washio, T., Dobbie, G., Huang, J.Z., Wang, R. (eds.) *PAKDD 2016. LNCS (LNAI)*, vol. 9652, pp. 491–502. Springer, Cham (2016). [https://doi.org/10.1007/978-3-319-31750-2\\_39](https://doi.org/10.1007/978-3-319-31750-2_39)
10. Song, D., Meyer, et al.: Recommending positive links in signed social networks by optimizing a generalized AUC. In: *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, pp. 290–296. AAAI Press (2015)
11. Liu, H., Wu, Z., Zhang, X.: CPLR: collaborative pairwise learning to rank for personalized recommendation. *Knowl.-Based Syst.* **148**, 31–40 (2018)
12. Rendle, S., Freudenthaler, et al.: BPR: Bayesian personalized ranking from implicit feedback. In: *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pp. 452–461. AUAI Press, Arlington (2009)
13. Kunegis, J., Gottron, T.: Online dating recommender systems: the split-complex number approach. In: *ACM Recsys Workshop on Recommender Systems and the Social Web*, pp. 37–44. ACM (2012)
14. Nakamura, A.: A UCB-Like strategy of collaborative filtering. In: *Proceedings of the Sixth Asian Conference on Machine Learning*, pp. 315–329. PMLR (2015)
15. Cai, X., Bain, M., et al.: Learning collaborative filtering and its application to people to people recommendation in social networks. In: *International Conference on Data Mining*, pp. 743–748. IEEE (2011)



# Robust Low-Rank Recovery with a Distance-Measure Structure for Face Recognition

Zhe Chen<sup>1</sup>, Xiao-Jun Wu<sup>1(✉)</sup>, He-Feng Yin<sup>1</sup>, and Josef Kittler<sup>2</sup>

<sup>1</sup> Jiangsu Province Engineering Laboratory of Pattern Recognition and Computational Intelligence, Jiangnan University, Wuxi 214122, China  
425493422@qq.com, xiaojun\_wu\_jnu@163.com, yinhefeng@126.com

<sup>2</sup> Centre for Vision, Speech and Signal Processing,  
University of Surrey, Guildford GU2 7XH, UK  
j.kittler@surrey.ac.uk

**Abstract.** Strict ‘0-1’ block-diagonal low-rank representation is known to extract more structured information. However, it is often overlooked that a test sample from one class may be well represented by the dictionary atoms from other classes. To alleviate this problem, we propose a robust low-rank recovery algorithm (RLRR) with a distance-measure structure (DMS) for face recognition. When representing a test sample, DMS highlights the energy of the low-rank coefficients when the distance from the corresponding dictionary atoms is small. Moreover, RLRR introduces a structure-preserving regularization term to strengthen the similarity of within-class coefficients. Besides, RLRR builds a link between training and test samples to ensure the consistency of representation. The alternative direction multipliers method (ADMM) is used to optimize the proposed RLRR algorithm. Experiments on three benchmark face databases verify the superiority of RLRR compared with state-of-the-art algorithms.

**Keywords:** Face recognition · Low-rank representation  
Distance-measure structure

## 1 Introduction

Sparse representation (SR) has been well studied for face recognition (FR) in the last few years. The original sparse representation based classification (SRC) [1] algorithm was proposed by Wright et al. Given an over-complete dictionary  $D \in R^{d \times K}$  and a query sample  $y \in R^d$ , where  $d$  and  $K$  denote sample dimensionality and dictionary size, respectively, the goal of SRC is to find a sparse linear representation  $\alpha \in R^K$  for  $y$  which has a few non-zero atoms by solving the following minimization problem:

$$\min_{\alpha} \|y - D\alpha\|_2^2 + \lambda \|\alpha\|_1 \quad (1)$$

However, SRC cannot perform well when there exist severe illumination or pose variations in face images. To deal with this problem, numerous SRC-based methods were presented [2, 3]. Based on the observation that collaborative mechanisms may be more important than sparsity, to this end, Zhang et al. [4] proposed a collaborative representation based classification (CRC) algorithm by utilizing  $l_2$  norm to replace  $l_1$  norm in SRC which can achieve comparable classification results in shorter time. In addition to SRC and CRC, Wang et al. [5] proposed a locality-constrained sparse coding (LLC) algorithm by embedding locality in the coding procedure. Unlike SRC, linear regression based classification (LRC) [6] used the training samples to represent a test sample class-wisely and classified it to the class which gives the minimum reconstruction error.

Low-rank recovery is another representation-based method which can capture more global structure information of samples rather than by imposing a low-rank constraint on representation. Robust principal component analysis (RPCA) [7] is a classical matrix recovery algorithm based on low-rank recovery theory and can be formulated as:

$$\min_{L,E} \|L\|_* + \lambda \|E\|_1 \quad s.t. \quad X = L + E \quad (2)$$

where  $X$  denotes the contaminated sample matrix,  $L$  and  $E$  denote the corresponding low-rank recovery and sparse error matrix, respectively.

Chen et al. [8] proposed a structured incoherence regularization term to encourage the independence of inter-class low-rank recoveries. Nevertheless, both [7] and [8] only considered the data come from a single subspace. In contrast, Liu et al. [9] presented a low-rank representation (LRR) method which assumes that the data are drawn from multiple subspaces, so LRR can be viewed as a generalized form of RPCA. Then, numerous LRR-based methods were proposed. For instance, LatLRR [10] was proposed to address the small sample size problem by using unobserved data to represent observed data. More recently, Zhou et al. [11] proposed an integrated low-rank-based discriminative feature learning algorithm (ILRDFL) by integrating feature learning and classification. Zhang et al. [12] proposed a structured low-rank recovery algorithm (SLRR) which can promote the similarity and coherence of within-class representations while weakening the negative influence of inter-class coefficients by imposing a block-diagonal structured regularization term on low-rank representation. By referring to the structured regularization term in SLRR. However, the strict ‘0-1’ block-diagonal structure in SLRR is skeptical because it is not reasonable to assume the within-class representations are the same (all ones) and that samples from other classes may not contribute to representing a test sample.

In this paper, we propose a novel robust low-rank representation algorithm (RLRR) with a distance-measure structure (DMS) to alleviate the negative influence caused by the above strict block-diagonal structure. The DMS strategy is different from the strict block-diagonal structure in SLRR, it judges which dictionary atoms are contribute to the low-rank representation according to the distance between the dictionary atoms and test samples. Concretely, the coefficients for which the corresponding dictionary atoms are away from the test samples

will be minimized, even if they belong to the same class. Besides, RLRR can preserve the representation’s whole structure by encouraging the similarity of within-class representations in a semi-supervised learning manner. Furthermore, inspired by Zhang’s [18] work, we build a link between training and test samples to promote the consistency of low-rank representation.

## 2 Robust Low-Rank Representation Algorithm (RLRR)

In this section, we propose a robust low-rank representation algorithm (RLRR) with a distance-measure structure (DMS). Note that  $X_{tr} = [X_{tr_1}, X_{tr_2}, \dots, X_{tr_c}] \in R^{d \times n}$  denotes the training samples and  $X_{tt} = [X_{tt_1}, X_{tt_2}, \dots, X_{tt_c}] \in R^{d \times m}$  denotes the test samples, where  $c$  is the number of class,  $n$  and  $m$  is the number of training and test samples, respectively. For the purpose of promoting the representation consistency, we define  $X = [X_{tr}, X_{tt}] \in R^{d \times (n+m)}$  (all samples) as the link matrix between  $X_{tr}$  and  $X_{tt}$ . Similar to LRR, we set original training samples  $X_{tr}$  as the dictionary, i.e.,  $D = X_{tr}$ . The objective function of proposed RLRR algorithm can be formulated as:

$$\begin{aligned} \min_{Z, E} \|Z\|_* + \alpha \|Z \odot \Phi\|_F^2 + \beta \|Z - \tilde{Z}\|_F^2 + \lambda \|E\|_1 \\ \text{s.t. } X = X_{tr}Z + E \end{aligned} \tag{3}$$

where  $\alpha$ ,  $\beta$  and  $\lambda$  are positive regularization parameters,  $Z = [Z_{tr}, Z_{tt}] \in R^{n \times (n+m)}$  denotes the representation matrix of sample  $X$  about dictionary  $X_{tr}$ .  $Z \odot \Phi$  denotes the Hardmard-Product of  $Z$  and  $\Phi$  and  $\Phi$  is the DMS matrix. In order to obtain  $\Phi$ , we first calculate the distance matrix  $A \in R^{n \times (n+m)}$  between dictionary  $X_{tr}$  and sample matrix  $X$ . Concretely, each  $A_{ij}$  ( $i = 1, 2, \dots, n, j = 1, 2, \dots, n + m$ ) can be obtained by calculating

$$A_{ij} = e^{-\frac{\|x_i - x_j\|^2}{\sigma}} \tag{4}$$

where  $\sigma$  is the heat kernel parameter,  $x_i$  and  $x_j$  are the sample vectors from  $X_{tr}$  and  $X$ , respectively. Then we choose the first  $\theta$  smallest elements to set them to 0 and the remaining  $K - \theta$  elements are revised as 1 column-wisely. After that, distance matrix  $A$  becomes an ‘0-1’ matrix  $\Phi$ . By minimizing  $\|Z \odot \Phi\|_F^2$ , the coefficients in  $Z$  which contribute to the representation in terms of dictionary atoms which are closer to the sample matrix  $X$  will be highlighted. In other words, when representing a sample from  $X$ ,  $\Phi$  enhances the robustness of the representation by weakening the negative effect caused by the dictionary atoms which are distant from the sample. Obviously,  $\Phi$  reflects a distance-measure structure rather than the ideal block-diagonal structure based on the assumption that any dictionary samples may contribute to representing a query sample even if they belong to different classes.

$\tilde{Z} = [\tilde{Z}_{tr}, \mathbf{0}] \in R^{n \times (n+m)}$  is of the same size as  $Z$  where  $\tilde{Z}_{tr} \in R^{n \times n}$  consists of the mean of the vectors of each class in  $Z_{tr}$ , thus the intra-class vectors in  $\tilde{Z}_{tr}$  are the same.  $\mathbf{0} \in R^{n \times m}$  indicates a matrix whose elements are all 0s because

the class information of test samples is unknown. By minimizing  $\|Z - \tilde{Z}\|_F^2$ , the similarity of within-class representations will be enhanced and the whole structure of representation  $Z$  can be preserved in a semi-supervised fashion.

### 3 The Optimization of RLR

We use the ADMM [13] algorithm to find the optimal low-rank representation. When we update a variable, the remaining variables are fixed in RLR. For convenience, we first introduce an auxiliary variable  $P$ , so formula (3) can be rewritten as:

$$\begin{aligned} \min_{Z, E, P} \|P\|_* + \alpha \|Z \odot \Phi\|_F^2 + \beta \|Z - \tilde{Z}\|_F^2 + \lambda \|E\|_1 \\ \text{s.t. } X = X_{tr}Z + E, Z = P \end{aligned} \quad (5)$$

then we obtain the Lagrangian function of problem (5):

$$\begin{aligned} L(Z, E, P, T_1, T_2, \mu) = \|P\|_* + \alpha \|Z \odot \Phi\|_F^2 + \beta \|Z - \tilde{Z}\|_F^2 \\ + \lambda \|E\|_1 + \frac{\mu}{2} \{\|X - X_{tr}Z - E + \frac{T_1}{\mu}\|_F^2 \\ + \|Z - P + \frac{T_2}{\mu}\|_F^2\} \end{aligned} \quad (6)$$

where  $T_1$  and  $T_2$  are Lagrangian multipliers,  $\mu$  is a positive penalty parameter.

**Updating  $Z$  with  $E, P$  Fixed:** Here, we define  $\Psi = I - \Phi$ , so  $Z \odot \Phi = Z \odot (I - \Psi) = Z - Z \odot \Psi$ . Let  $\Omega = Z \odot \Psi$ , thus  $Z \odot \Phi = Z - \Omega$ . In formula (6),  $Z$  has the following closed-form solution:

$$\begin{aligned} Z_{k+1} = [(2\alpha + 2\beta + \mu_k)I + \mu X_{tr}^T X_{tr}]^{-1} (2\alpha\Omega + 2\beta\tilde{Z} \\ + \mu_k X_{tr}^T X - \mu_k X_{tr}^T E_k + X_{tr}^T T_1^k + \mu_k P_k - T_2^k) \end{aligned} \quad (7)$$

**Updating  $P$  with  $Z, E$  Fixed:**

$$P_{k+1} = \arg \min_P \frac{1}{\mu} \|P\|_* + \frac{1}{2} \|P - (Z_{k+1} + \frac{T_1^k}{\mu_k})\|_F^2 \quad (8)$$

Formula (8) can be solved by the singular value thresholding operator algorithm [14].

**Updating  $E$  with  $Z, P$  Fixed:**

$$E_{k+1} = \arg \min_E \frac{\lambda}{\mu_k} \|E\|_1 + \frac{1}{2} \|E - (X - X_{tr}Z_{k+1} + \frac{T_2^k}{\mu_k})\|_F^2 \quad (9)$$

Formula (9) can be solved by the soft-thresholding operator [15]. After we optimize  $Z, P$  and  $E$ , the ADMM also needs to update the Lagrange multipliers  $T_1$  and  $T_2$ , as well as  $\mu$ , for faster convergence.

## 4 Classification Method

We can obtain an optimized low-rank representation  $\hat{Z} = [Z_{tr}, Z_{tt}]$ , where  $Z_{tr} \in R^{n \times n}$  and  $Z_{tt} \in R^{n \times m}$  are the sub-representations corresponding to  $X_{tr}$  and  $X_{tt}$ , respectively. Given a label matrix  $H \in R^{c \times n}$  of training samples  $X_{tr}$ , a linear classifier with weight  $W$  can be learned by the following objective function:

$$\hat{W} = \arg \min_W \|H - WZ_{tr}\|_F^2 + \gamma \|W\|_F^2 \quad (10)$$

where  $\gamma$  is a positive parameter. Formula (10) is a Ridge Regression-based model and  $W$  has the closed-form solution:

$$\hat{W} = HZ_{tr}^T(Z_{tr}Z_{tr}^T + \gamma I)^{-1} \quad (11)$$

When given a test sample  $X_{tt}^i \in X_{tt}$ , we can obtain its classification result by:

$$\textit{identity}(X_{tt}^i) = \arg \max_j (\hat{W}Z_{tt}^i) \quad (12)$$

where  $Z_{tt}^i$  is the  $i$ th column of  $Z_{tt}$  which corresponds to sample  $X_{tt}^i$ .

## 5 Experiments

In this section, we conduct experiments on three benchmark face databases: AR [16], Extended Yale B [17] and Labeled Faces in the wild (LFW) [19]. In the experiments, SRC [1], CRC [4], LRSI [8], LLC [5], SLRR [12], LatLRR [10], RPCA [7] and ILRDFL [11] algorithms are compared with our RLRR algorithm. For fairness, LatLRR and RPCA use the same classification method with our RLRR. SRC, CRC and LRSI use all training samples as the dictionary. For all databases, we set the number of neighbors of LLC as 5. On AR and Extended Yale B databases, the parameters are set to  $\alpha = 1$ ,  $\beta = 0.5$  and  $\gamma = 0.5$  and on LFW database  $\alpha = 1$ ,  $\beta = 0.5$  and  $\gamma = 0.1$  as determined by cross-validation method. The values of  $\theta$  are set to 150, 350 and 100 in AR, LFW and Extended Yale B databases, respectively.

### 5.1 Experiments on AR Database

AR face database consists of 4000 images of 126 individuals (70 men and 56 women), each image has a size of  $165 \times 120$  pixels. For convenience, we choose a subset of AR database which has 2600 face images of 100 individuals (50 men and 50 women), each individual has 26 images which can be separated into two sessions and each session has 13 images (7 with varying illumination conditions and expressions, 3 with sunglasses and 3 with scarf). In our experiments, all images are normalized and resized to  $55 \times 40$  pixels. In SLRR, the dictionary's size is set to 500 which has been reported to be the optimal size in ILRDFL [11]. Following SLRR [12], on AR database, we also consider three scenarios: Sunglasses, Scarf and Mixed. Because there exists 3 images with sunglasses or



scarf for training in each individual, we repeat experiments three times in each scenario, then calculate the average as the final results. The recognition results of different algorithms are reported in Table 1. From Table 1, we see our RLRR algorithm achieves better recognition performance in different scenarios.

**Table 1.** Recognition rates (%) of different algorithms on the AR database

Scenario	Sunglasses	Scarf	Mixed
SRC [1]	88.6	85.6	83.2
CRC [4]	90.0	87.1	86.9
LLC [5]	87.1	85.8	84.1
LRSI [8]	84.7	78.6	81.3
RPCA [7]	88.1	86.6	86.5
SLRR [12]	89.0	85.3	84.8
LatLRR [10]	87.1	86.3	84.3
ILRDFL [11]	90.9	91.8	91.2
<b>RLRR</b>	<b>94.2</b>	<b>93.1</b>	<b>93.2</b>

## 5.2 Experiments on LFW Database

The LFW database consists of images of 5,749 individuals and we use the subset LFW-a to conduct experiments. LFW-a is an aligned version of LFW based on commercial face alignment software. We use the subjects that include no less than ten samples and we construct dataset with 158 subjects from LFW-a. For each subject, we randomly chose 5 images for training and 5 images for testing. So there are 790 training samples and 790 test samples. All samples are normalized and resized to  $90 \times 90$  pixels. The dictionary size of SLRR is set to 790. The classification results of different algorithms are shown in Table 2. From Table 2, we can find that the proposed RLRR method obtains better classification performance than other low-rank based methods, i.e., SLRR, LatLRR and ILRDFL.

## 5.3 Experiments on Extended Yale B Database

The Extended Yale B database consists of about 2414 face images of 38 individuals, each image has been cropped to  $196 \times 128$  pixels. In our experiments, all images are resized to  $48 \times 42$  pixels and the dictionary size is set to 190 for SLRR. For each person, we randomly select 10, 15, 20 and 25 images as training samples and the remaining images are used as test samples. The average recognition accuracy of 10 runs are shown in Table 3. From Table 3, we observe that our RLRR is superior to other algorithms in all conditions.

**Table 2.** Recognition rates (%) of different algorithms on the LFW database

Algorithms	Recognition rates
SRC [1]	68.4
CRC [4]	64.7
LLC [5]	60.1
SLRR [12]	68.2
LRRC [22]	62.2
LatLRR [10]	52.7
ILRDFL [11]	55.3
<b>RLRR</b>	<b>72.3</b>

**Table 3.** Recognition rates (%) of different algorithms on the Extended Yale B database

Training samples per person	10	15	20	25
SRC [1]	87.9	93.6	96.4	98.0
CRC [4]	84.4	91.7	95.7	97.3
LLC [5]	77.8	87.0	90.9	93.9
LRSI [6]	87.7	92.3	94.6	96.4
RPCA [7]	86.2	91.3	94.1	95.9
SLRR [12]	85.2	92.2	94.8	96.6
LatLRR [10]	83.3	88.8	92.3	94.6
ILRDFL [11]	88.3	94.2	96.4	98.7
<b>RLRR</b>	<b>89.7</b>	<b>95.3</b>	<b>97.6</b>	<b>99.1</b>

## 6 Conclusions

In this paper, we propose a robust low-rank representation (RLRR) algorithm with a distance-measure structure (DMS) for face recognition. We first introduce a distance-measure structure which is different from the ideal block-diagonal structure involved in SLRR. The DMS strategy judges which coefficients can contribute to the representation by calculating a distance matrix between dictionary atoms and test samples. Second, by constructing a link for training and test samples, a structure-preserving regularization term is proposed to encourage the coherence of intra-class representation in a supervised way and the consistency of representation also can be promoted. Experiments on three benchmark face databases demonstrate the effectiveness of the proposed RLRR algorithm.

## References

1. Wright, J., Yang, A.Y., Ganesh, A., Sastry, S.S., Ma, Y.: Robust face recognition via sparse representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **31**(2), 210–227 (2009)
2. Xu, Y., Zhang, D., Yang, J., Yang, J.Y.: A two-phase test sample sparse representation method for use with face recognition. *IEEE Trans. Circuits Sys. Video Technol.* **21**(9), 1255–1262 (2011)
3. Lu, C., Min, H., Gui, J., Zhu, L., Lei, Y.: Face recognition via weighted sparse representation. *J. Vis. Commun. Image Represent.* **24**(2), 111–116 (2013)
4. Zhang, L., Yang, M., Feng, X.C.: Sparse representation or collaborative representation which helps face recognition? In: *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 471–478 (2011)
5. Wang, J., Yang, J., Yu, K., Lv, F., Huang, T., Gong, Y.: Locality-constrained linear coding for image classification. In: *Proceedings of the 23rd IEEE Conference on Computer Vision and Pattern Recognition*, San Francisco, CA, USA, pp. 3360–3367, June 2010
6. Naseem, I., Togneri, R., Bennamoun, M.: Linear regression for face recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **32**(11), 2106–2112 (2010)
7. Candès, E.J., Li, X., Ma, Y., Wright, J.: Robust principal component analysis? *J. ACM* **58**(3), Art. ID 11 (2011)
8. Chen, C.-F., Wei, C.-P., Wang, Y.-C.F.: Low-rank matrix recovery with structural incoherence for robust face recognition. In: *Proceedings of 25th IEEE Conference on Computer Vision and Pattern Recognition*, Providence, RI, USA, pp. 2618–2625, June 2012
9. Liu, G., Lin, Z., Yu, Y.: Robust subspace segmentation by low-rank representation. In: *ICML* (2010)
10. Liu, G., Yan, S.: Latent low-rank representation for subspace segmentation and feature extraction. In: *Proceedings of the 13th International Conference on Computer Vision*, Barcelona, Spain, pp. 1615–1622, November 2011
11. Zhou, P., Lin, Z., Zhang, C.: Integrated low-rank-based discriminative feature learning for recognition. *IEEE Trans. Neural Netw. Learn. Syst.* **27**(5), 1080–1093 (2016)
12. Zhang, Y., Jiang, Z., Davis, L.S.: Learning structured low-rank representations for image classification. In: *Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition*, Portland, OR, USA, pp. 676–683, June 2013
13. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J.: Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.* **3**(1), 1122 (2011)
14. Cai, J., Candès, E., Shen, Z.: A singular value thresholding algorithm for matrix completion. *SIAM J. Optim.* **20**(4), 1956–1982 (2010)
15. Lin, Z., Chen, M., Ma, Y.: The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. *arXiv preprint arXiv:1009.5055* (2010)
16. Martinez, A.M., Benavente, R.: The AR face database, Computer Vision Center, Barcelona, Spain, Technical report #24, June 1998
17. Lee, K.C., Ho, J., Driegman, D.: Acquiring linear subspaces for face recognition under variable lighting. *IEEE Trans. Pattern Anal. Mach. Intell.* **27**(5), 684–698 (2005)

18. Zhang, Z., Xu, Y., Shao, L., Yang, J.: Discriminative block-diagonal representation learning for image recognition. *IEEE Trans. Neural Netw. Learn. Syst.* (2017). <https://doi.org/10.1109/TNNLS.2017.2712801>
19. Huang, G.B., Ramesh, M., Berg, T., Learned-Miller, E.: Labeled faces in the wild: a database for studying face recognition in unconstrained environments, Univ. Massachusetts Amherst, Amherst, MA, USA, Technical report UM-CS-2014-003, October 2007



# A Surface Defect Detection Method Based on Positive Samples

Zhixuan Zhao<sup>1</sup>, Bo Li<sup>1(✉)</sup>, Rong Dong<sup>2</sup>, and Peng Zhao<sup>2</sup>

<sup>1</sup> School of Electronic Science and Engineering, Nanjing University, Nanjing, China  
liboee@nju.edu.cn

<sup>2</sup> Nanjing Huichuan Image Vision Technology Co. Ltd., Nanjing, China

**Abstract.** Surface defect detection and classification based on machine vision can greatly improve the efficiency of industrial production. With enough labeled images, defect detection methods based on convolution neural network have achieved the detection effect of state-of-art. However in practical applications, the defect samples or negative samples are usually difficult to be collected beforehand and manual labelling is time-consuming. In this paper, a novel defect detection framework only based on training of positive samples is proposed. The basic detection concept is to establish a reconstruction network which can repair defect areas in the samples if they are existed, and then make a comparison between the input sample and the restored one to indicate the accurate defect areas. We combine GAN and autoencoder for defect image reconstruction and use LBP for image local contrast to detect defects. In the training process of the algorithm, only positive samples is needed, without defect samples and manual label. This paper carries out verification experiments for concentrated fabric images and the dataset of DAGM 2007. Experiments show that the proposed GAN+LBP algorithm and supervised training algorithm with sufficient training samples have fairly high detection accuracy. Because of its unsupervised characteristics, it has higher practical application value.

**Keywords:** Positive samples · Surface defect detection · Autoencoder · GAN

## 1 Introduction

Surface defect detection plays a very important part in industrial production process. It is of significant impact on the quality and reputation of the final products in the market. Traditionally, surface defects are inspected by human vision, which is subjective, costly, inefficient and inaccurate.

Machine vision system is a possible substitution of human vision, but it also encounters many problems and challenges in practical applications, especially during those years when traditional image features used to discriminate defects and non-defects are designed manually based on experience. The characteristics of traditional image feature extraction operators are usually at low level. In the case of complex scene variations such as illumination change, perspective distortion, occlusion, object deformation and so on, the extracted features are often not robust enough to handle them so that many

algorithms are not applicable in practical contexts. Recently, deep learning has been demonstrated to be very powerful in the extraction of image features. The convolution neural network has achieved the highest precision in all kinds of supervised problems, such as classification, target location, semantic segmentation and so on.

Faghih-Roohi et al. [1] uses deep convolution neural network to perform defect detection on rail surface. It divides rail images into 6 categories, including 1 category of non-defect images and 5 categories of defect images, and then DCNN is used to classify them; Liu et al. [2] proposed a two-stage method which combines the region proposals by selective search and the convolution neural network. It detects and identifies the obtained regions, and then completes the detection of the surface defects of the capsule; Yu et al. [3] uses two FCN [4] semantic segmentation networks to detect defects. One of them is coarse positioned, and another one is fine positioned. It can accurately draw the outline of defects, and has achieved higher accuracy than the original FCN on the dataset of DAGM 2007 [12] and can be completed in real time.

All of the above algorithms has used supervised schemes to detect defects. Two problems are necessary to be considered in the practical application of industrial detection:

**Lack of Defect/Negative Samples in Training Samples.** In practical problems, there are always fewer defects in the training samples because it is hard to collect many defect samples beforehand. Therefore, the number of positive and negative samples in the training process is extremely unbalance so that the generated model may be unstable or invalid. In the scene where the defect appearance is variable and unpredictable, supervised detection methods often fail to reach the required precision.

**Manual Labelling is Expensive.** In the actual defect detection applications, there are usually many different kinds of defects, and detection standards and quality index are often different. This requires a large number of training samples to be manually label for specific needs, which needs so much human resources.

In view of the problems existing in the practical application of the above supervised learning algorithm, a defect detection method based on positive sample training is proposed. The training process only needs to provide sufficient positive samples, without the need to provide defect samples, and without manual labeling, the effect of defect detection can be achieved.

## 2 Related Work

### 2.1 Defect Repair Model Based on Positive Samples

The inspiration for the model we have proposed comes from a series of GAN [5] based repair and detection models. As shown in the Fig. 1 is the schematic diagram of the GAN principle. The generator  $G$  receives a Gaussian random signal to generate a picture, the discriminator  $D$  receives a true or false picture, and outputs the probability of the picture is true. The reality degree of the generated picture will be improved in the continuous game of the generator and the discriminator.

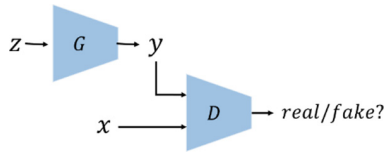


Fig. 1. The architecture of GAN

Yeh et al. [6] uses GAN for image repair. First, it uses a no defect picture to train a GAN model in a general process. Then, when repairing a known location defect, we optimize the input  $z$  of the generator  $G$ , so that we can find the best  $z$ , which makes  $y$  and the normal part of a defective picture similar to the greatest extent. The picture  $y$  is the restored image. Schlegl et al. [7] implements defect detection on the basis of image repair. First, it uses the reconstruction error of the middle layers to complete a repair module without knowing the location of the defect in advance. Second, it makes the difference between the restored picture and the original one. Where the difference is large, that is the defect. Because of the error of reconstruction and repair, the disadvantage of this model is that it is difficult to distinguish the reconfiguration error and the small defect by the direct subtraction.

The obvious disadvantage of these two models is that they use gradient optimization to find the right  $z$ , and then get the repair picture further. This process needs to consume a lot of time, which is so unpractical. So we expect that using autoencoder to restore the defect image.

## 2.2 Autoencoder

Pix2pix [8] uses autoencoder to cooperate with GAN to solve the task of image translation. It can generate sharp, realistic images. In order to achieve better results in details and edge parts, pix2pix uses the structure of the skip connections, like Unet [9]. This structure is not suitable for removing the whole defect, so it is not used in our model. The general image translation task refers to the task of coloring black and white pictures, translating simple strokes into photographs, and so on. We use a similar structure to achieve the transformation of the defect picture to the restored picture.

On the basis of the above research, the following work has been completed in this paper: (1) We use the autoencoder to restore the image. We can complete the image repair function in real-time and improve the quality of the picture with GAN loss; (2) We use artificial defects in training, and we do not rely on a large number of full and real defect samples and manual label; (3) We use LBP [10] to compare the restored image and the original one to find the location of the defect more accurately.

To sum up, we propose a defect detection model, which is based on the training of the positive example without manual label.

### 3 Method

The general framework of the model presented in this article is shown in the Fig. 2. At the training stage,  $x$  is a random picture taken randomly from the training set.  $C(x \sim |x)$  is an artificial defect module. Its function is to automatically generate a damaged, defective sample.  $x \sim$  is its output. EN and DE constitute an autoencoder  $G$ . EN is an encoder, and DE is a decoder, and the entire autoencoder can be seen as a generator in the GAN model. The task of  $G$  is to fix a defective picture.  $D$  is a discriminator, and the output of  $D$  is the probability that its discriminant is a true positive sample.

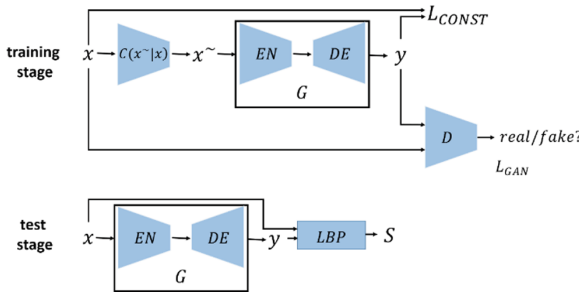


Fig. 2. The framework of our model

At the test stage, we input the test picture  $x$  into the autoencoder  $G$ , get the restored image  $y$ . Then use the LBP algorithm to extract the features of  $x$  and  $y$ , and compare the features of each pixel of  $x$ , where the feature difference between  $x$  and  $y$  is large, that is the defect.

#### 3.1 Objective

The samples with defects should be equal to the original positive samples after being autoencoder. Here we refer to pix2pix using L1 distance as a similar basis for them. L1 distance preferred less blurred images than L2 distance. The refactoring error is defined here:

$$L_{CONST}(G) = E_{x \sim P_{data}(x)} [\|x - G(x \sim)\|_1] \tag{1}$$

If the reconfiguration error is used only as the target function, the edges of the obtained images are blurred and the details are lost. According to the experiment in pix2pix, a discriminant network is introduced and GAN loss can be added to improve the image blurred problem and enhance the fidelity of the image. The objective of a GAN can be expressed as:

$$L_{GAN}(G,D) = E_{x \sim P_{data}(x)} [\log D(x) + \log(1 - D(G(x \sim)))] \tag{2}$$



So the overall optimization goal is to find the parameters that generate the network  $G$ , and make it satisfied:

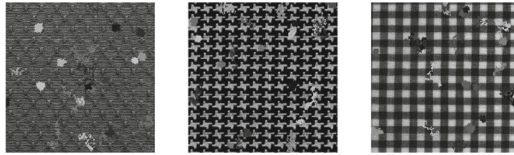
$$G^* = \arg \min_G \max_D (L_{GAN}(G,D) + \lambda L_{CONST}(G)) \quad (3)$$

$\lambda$  is the parameter to balance the GAN loss and the reconstruction error, which is determined by the experiment. The introduction of GAN loss, to some extent, will compete with the refactoring error, but it can improve the quality of the picture and the description of the important details.

### 3.2 Network Structure and Artificial Defects

The network structure of our proposed model is referred to as DCGAN [11]. In the generator and discriminator network, batchnorm layer is added. The LeakyRelu layer is used in the discriminator network, and the Relu layer is used in the generator network. The encoder structure is roughly similar to the discriminator.

In our model, the autoencoder only needs to repair the original map to the nearest example sample, which does not need to know the specific form of the defect. So the network will be able to learn the information of the repair map when enough random defects are attached to the sample. In actual training, we manually generate random blocks, locations, sizes, grayscale values, and the number of defect blocks added to the picture, as shown in Fig. 3, training network to automatically repair defects.



**Fig. 3.** Artificial defect schematic diagram

For data augmentation, we adopt random resize between 0.5 and 2, and additionally add random rotation between  $-180$  and  $180^\circ$ , and random Gaussian blur for the images.

### 3.3 To Get the Position of the Defect

Because there are some errors in the detail information of the restored picture, we should not directly divide the restored picture and the original picture to get the position of the defect directly. We use the LBP [10] algorithm for feature extraction, and then search for the most matched pixels around each pixel. The LBP algorithm is a nonparametric algorithm which has the characteristics of light invariance and is suitable for the dense points.

The steps to get the defective picture as shown in the Fig. 4. The original picture  $x$  and the restored picture  $y$  are processed by LBP algorithm to get the feature map  $x^+$  and  $y^+$ . For each pixel point of the  $x^+$ , search the nearest eigenvalue point at the corresponding location of the  $y^+$ , which is the point of the pixel as the matching point. Make the

difference between the eigenvalues of the two matching points and get the absolute value. The smaller the value you get, the lower the possibility that the point is a defect. Then using the fixed threshold binaryzation, you can get to the position of the defect.

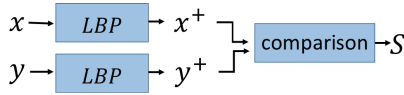


Fig. 4. The process of getting the defect position

## 4 Experiment

### 4.1 Preparation

This paper uses the fabric picture and the texture surface picture to test the performance of the experimental model. There are 3 kinds of fabric pictures and 1 kind of texture surface pictures. The image of the fabric comes from the database [13], and the texture surface image is from the dataset of DAGM 2007 [12]. In this paper, we compare the supervised semantic segmentation model [4] and the proposed model in defect detection.

The develop environment is as follows: CPU: Intel® Xeon(R) E5620@2.40GHZ\*16, GPU: GTX1080, memory: 16 G, python 2.7.12 and mxnet. We train by Adam, and set the initial learning rate to 0.0002, and set the batch size to 64.

### 4.2 Result

We use the average accuracy of the whole picture as the evaluation index of the model performance in the experiment.

**Texture Surface.** Texture surface has good consistency, so they have enough defect samples on the training set to learn (Tables 1 and 2).

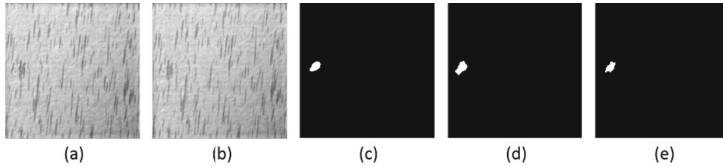
Table 1. Test information of texture surface

Training set	400 images without defects for ours 85 (defective) + 400 (no defect) for FCN
Test set	85 images with defects
Picture size	512*512

**Table 2.** Test result of texture surface

Model	Mean accuracy	Time cost
FCN(8s)	98.3547%	80.3 ms
Ours	98.5323%	52.1 ms

As shown in Fig. 5, it is an example of some defect detection results.



**Fig. 5.** (a) Initial input images. (b) Restored images (c) Results of ours. (d) Results of FCN. (e) Ground truth.

**Fabric Picture.** Due to the different form of fabric samples in real scenes, the defect samples in training set are relatively scarce. In this experiment, there are 5 types of defects. There are 5 pictures in each form, and 25 positive pictures. For the supervised semantic segmentation model, 3 of each form of defect picture is used as a training set, and 2 is used as a test set (Tables 3 and 4).

**Table 3.** Test information of fabric picture

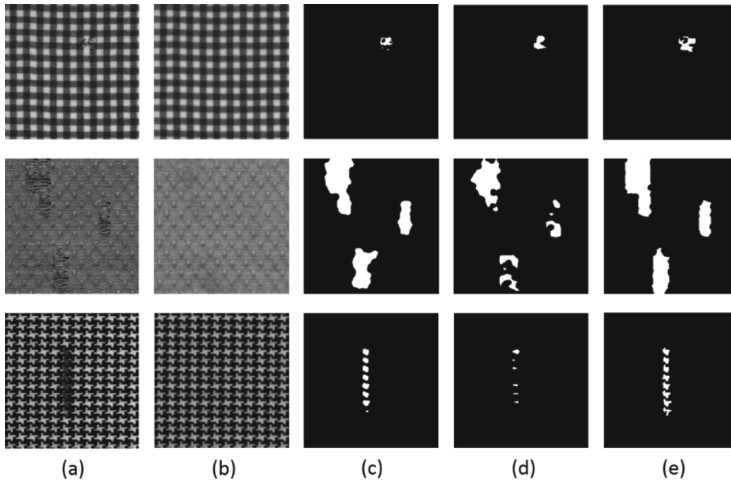
Training set	75 images without defects for ours 45 (defective) + 75(no defect) for FCN
Test set	30 images with defects
Picture size	256*256

**Table 4.** Test result of fabric picture

Model	Mean accuracy	Time cost
FCN(8s)	81.6833%	31.2 ms
Ours	94.4253%	22.3 ms

As shown in Fig. 6, it is an example of some defect detection results.

Experiments show that in a regular pattern background, our model can obtain and supervised semantic segmentation accuracy when the labeled defect samples are sufficient, and our model can obtain higher precision when the defective sample with annotation is not sufficient. In terms of time consumption, our model can achieve real-time detections.



**Fig. 6.** (a) Initial input images. (b) Restored images (c) Results of ours. (d) Results of FCN. (e) Ground truth.

## 5 Conclusion

In this paper, we combine autoencoder and GAN to propose a defect detection model based on positive sample training without manual label. In training, combined with artificial defects and data enhancement methods, the model can automatically repair the defects of regular pattern texture images, and get the specific location of defects through comparing the features of the original picture and the restored picture. The position of the defect can be detected in real time on the image of the fabric and the plane of the texture. Moreover, we can get better results than supervised semantic segmentation when training defect instances are scarce.

If the background is too complex and random, it is difficult for the autoencoder to reconstruct and repair the picture. The related defect detection problem remains to be studied in the future.

**Acknowledgments.** This work is partially supported by the Key Project supported by Shenzhen Joint Funds of the National Natural Science Foundation of China (Gran No. U1613217).

## References

1. Faghih-Roohi, S., et al.: Deep convolutional neural networks for detection of rail surface defects. In: International Joint Conference on Neural Networks. IEEE (2016)
2. Liu, R., et al.: Region-convolutional neural network for detecting capsule surface defects. *Boletín Técnico* **55**(3), 92–100 (2017)
3. Yu, Z., Wu, X., Gu, X.: Fully Convolutional networks for surface defect inspection in industrial environment. In: Liu, M., Chen, H., Vincze, M. (eds.) ICVS 2017. LNCS, vol. 10528, pp. 417–426. Springer, Cham (2017). [https://doi.org/10.1007/978-3-319-68345-4\\_37](https://doi.org/10.1007/978-3-319-68345-4_37)

4. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431–3440 (2015)
5. Goodfellow, I., et al.: Generative adversarial nets. In: Advances in Neural Information Processing Systems (2014)
6. Yeh, R., et al.: Semantic image inpainting with perceptual and contextual losses (2016). arXiv preprint [arXiv:1607.07539](https://arxiv.org/abs/1607.07539)
7. Schlegl, T., Seeböck, P., Waldstein, Sebastian M., Schmidt-Erfurth, U., Langs, G.: Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In: Niethammer, M., Styner, M., Aylward, S., Zhu, H., Oguz, I., Yap, P.-T., Shen, D. (eds.) IPMI 2017. LNCS, vol. 10265, pp. 146–157. Springer, Cham (2017). [https://doi.org/10.1007/978-3-319-59050-9\\_12](https://doi.org/10.1007/978-3-319-59050-9_12)
8. Isola, P., et al.: Image-to-image translation with conditional adversarial networks (2017). arXiv preprint
9. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). [https://doi.org/10.1007/978-3-319-24574-4\\_28](https://doi.org/10.1007/978-3-319-24574-4_28)
10. Ojala, T., Pietikäinen, M., Harwood, D.: A comparative study of texture measures with classification based on featured distributions. *Pattern Recogn.* **29**(1), 51–596 (1996)
11. Radford, A., Luke M., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks (2015). arXiv preprint [arXiv:1511.06434](https://arxiv.org/abs/1511.06434)
12. HCI: Weakly Supervised Learning for Industrial Optical Inspection. <https://hci.iwr.uni-heidelberg.de/node/3616>. Accessed 13 Nov 2017
13. Ngan, H.Y.T., Pang, G.K.H., Yung, N.H.C.: Automated fabric defect detection—a review. *Image Vis. Comput.* **29**(7), 442–458 (2011)



# Joint Multi-field Siamese Recurrent Neural Network for Entity Resolution

Yang Lv, Lei Qi, Jing Huo, Hao Wang, and Yang Gao<sup>✉</sup>

State Key Laboratory for Novel Software Technology,  
Nanjing University, Nanjing, China  
gaoy@nju.edu.cn

**Abstract.** Entity resolution which deals with determining whether two records refer to the same entity has a wide range of applications in both data cleaning and integration. Traditional approaches focus on using string metrics to calculate the matching scores of recorded pairs or employing the machine learning technique with hand-crafted features. However, the effectiveness of these methods largely depends on designing good domain-specific metric methods or extracting discriminative features with rich domain knowledge. Also, traditional learning-based methods usually ignore the discrepancy between citation's fields. In this paper, to decrease the impact of information gaps between different fields and fully take advantage of semantical and contextual information in each field, we present a novel joint multi-field siamese recurrent architecture. In particular, our method employs word-based Long Short-Term Memory (LSTM) for the fields with the strong relevance between each word and character-based Recurrent Neural Network (RNN) for the fields with the weak relevance between each word, which can exploit each field's temporal information effectively. Experimental results on three datasets demonstrate that our model can learn discriminative features and outperforms several baseline methods and other RNN-based methods.

**Keywords:** Entity resolution · Joint multi-field siamese architecture  
Recurrent Neural Network · Long Short-Term Memory

## 1 Introduction

Entity resolution is an important technique for both accurate analysis and cost-effectiveness [7] in data cleaning and integration. An entity is an object in real world such as a citation, a person, a product, etc. However, due to spelling mistakes or sometimes the use of abbreviations, there usually exist some errors or confusions in these entity data, which make accurate retrieval of the records of the same entity difficult.

In recent years, many efforts have been done for entity resolution. Generally, these methods utilize weighted string metrics to judge whether two records refer to the same entity [1]. Indeed, in some specific tasks, these methods have a good

performance by employing well-designed metrics. However, designing an effective string metric is difficult, which needs much domain experience. In addition, some other methods employ labeled data and consider entity resolution as a binary classification problem [5]. They need to manually construct features from text records, which are used to train a classification model such as Support Vector Machine. Unlike weighted string metrics, these methods are metric-free and can automatically learn classification model from labeled data. However, their performance depends heavily on the effectiveness of hand-crafted features.

**Table 1.** Two citations from Citeseer dataset.

Author	Title	Venue
S. Mahadevan, J. Connell	Automatic programming of behavior-based robots using reinforcement learning	Artificial Intelligence
Mahadevan S., Connel J	Automatic programming of behavior-based robots using reinforcement learning	In: Artificial Intelligence

In this paper, to realize an end-to-end learning-based method which can automatically extract recorded pairs’ semantical features for entity resolution in the citation domain, we propose a novel joint multi-field siamese recurrent neural network. Generally, one citation consists of author, title and venue. Table 1 shows two citations sampled from Citeseer dataset. We can observe that there are no obvious relationships between author, title and venue, that is, there are information gaps between citation fields. Therefore, we treat each field as a sub-sentence and use one RNN network to capture its contextual semantics and learn their discriminative features, then using a full-connected layer to fuse all these features from each field as the feature of one citation. Especially, since the words in the title field have a strong relevance and a weak relevance exists between the words in the author/venue field, word-based LSTM and character-based RNN are employed for extracting title and author/venue features, respectively. The major contributions of this paper can be summarized as follows:

- We propose a novel joint multi-field siamese recurrent neural network, which can fuse the multi-field features effectively.
- Word-based LSTM and character-based RNN are employed for extracting title and author/venue features, respectively, which can fully exploit the contextual information of different fields.
- Comparison with several baseline approaches and other RNN-based methods, our method achieves better performance on two public datasets and one synthetic dataset.

## 2 Related Work

Currently, there has been a great deal of work on entity resolution [4, 7, 8]. In this section, we give a brief review of most related work.

There has existed lots of metric-based methods for entity resolution [1, 2, 6]. Firstly, these methods applied the string metric to get similarity scores for each record pair’s field. Then, the strategies (i.e., the summed weights) were employed to fuse all of these scores from different fields to obtain the record pair’s similarity score. Finally, a decision was made by comparing the score with the predefined threshold. However, for these methods, designing a proper metric method and choosing weights of different fields need the domain-specific experience.

In addition, entity resolution can be considered as a learning problem. In [3], one learnable text similarity measure was proposed. Record pairs were converted to feature vectors and one machine learning model like Support Vector Machine was trained for classification. Minton *et al.* predefined domain-specific rules [11] for entity resolution. According to these rules, the process of record pair’s field matching can be converted into an ordered chain of activated rules. By employing naive Bayes classifier to the chain, the similarity of the record pair can be obtained. Since these methods extract features manually, it is difficult to obtain the semantical and contextual information from the text record.

Recently, deep learning has been applied for many tasks successfully, such as computer vision and natural language processing. Also, several methods based on RNN have been proposed for entity resolution [12, 13] with the single field. In [12], a siamese recurrent architecture was proposed for learning sentence similarity, which consists of two word-level LSTM networks. Neculoiu *et al.* employed analogous architecture to job title normalization, which is composed of four character-based bidirectional LSTM networks [13]. However, due to the gaps between different fields in one entity such as citation, using these methods directly is not suitable. Therefore, we propose a novel joint multi-field siamese recurrent architecture for entity resolution, which can effectively decrease the impact of information gaps between different fields and fully take advantage of semantical and contextual information in each field.

## 3 Our Method

### 3.1 Problem Definition

In this paper, we consider entity resolution as a supervised learning task and choose entity resolution on citations with labeled training data. Table 1 shows a pair of citations from Citeseer dataset. Each citation consists of three fields including author, title and venue. If two citations are of the same entity, the label is 1, otherwise, 0. Formally, training data is defined as  $R = \{r_1, \dots, r_n\}$ , where  $r_k = (c_i, c_j, y_k)$ ,  $r_k$  represents the  $k^{th}$  training pair with label  $y_k \in \{0, 1\}$  and  $c_i/c_j$  denotes  $i^{th}/j^{th}$  citation.



### 3.2 Joint Multi-field Siamese Recurrent Neural Network

Generally, since each citation can be naturally divided into multiple fields such as author, title and venue, each citation is not a consistent text. Information gaps exist between two neighboring fields. In addition, different fields have different characteristics. For example, the title field could be view as short sentence with certain local contextual relationship. The author and venue fields have only several words without obvious relevance. Considering these characteristics above, we propose a novel joint multi-field siamese recurrent neural network which is illustrated in Fig. 1.

As is shown in Fig. 1, our siamese network can be divided into three parts (two symmetrical subnets and one loss layer). Each subnet is made up of several RNNs. The input of each RNN is the fields of citations. Considering the characteristics of different fields, LSTM and conventional RNN are employed for title and author/venue fields, respectively. All of their final hidden states will be concatenated together and the concatenated vector is passed to a full-connected layer to get the discriminative vector for the citation. In addition, the contrastive loss [9] is chosen as the loss function in the proposed network, which optimizes the weights of the deep network to make features of the same entity more similar and have a margin for different entities. In the testing phase, if the cosine similarity of two citations exceeds 0.5, they are matched.

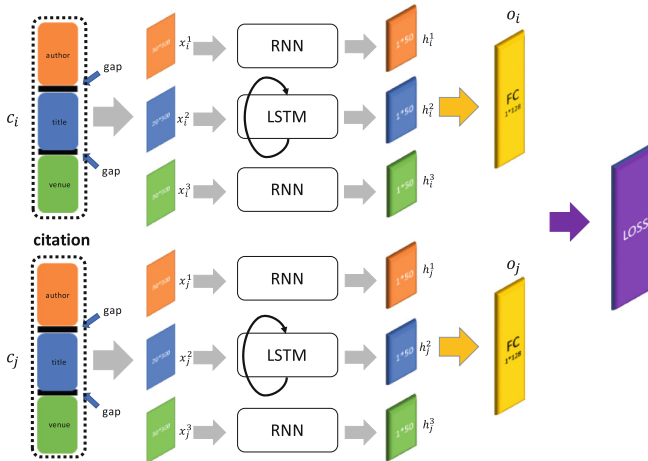
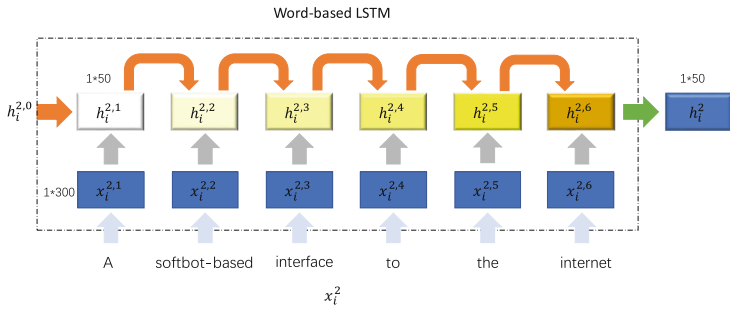


Fig. 1. Overview of the joint multi-field siamese recurrent neural network.

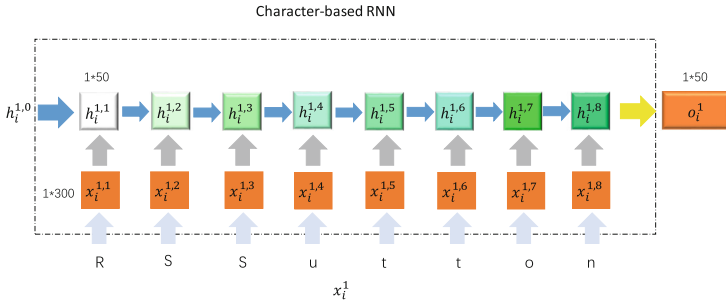
**Word-Based LSTM.** The title field is the main part in citation which contains contextual information. Just like a semantically continuous sentence, the previous one or several words in the title field may be related to the next words. Considering the characteristic, we split the title field into the word list and keep their original order. Specially, the internal gating mechanisms of LSTM

can regulate the propagation of certain relevant contexts, which can enhance the discriminative ability of local features and the memory unit which has an internal recurrence can effectively address the vanishing gradient problem. Thus, we choose LSTM for dealing with the title field, which is shown in Fig. 2. After the title field is divided into several words, LSTM processes each word sequentially. When all words are processed, the final hidden state is provided for next step. In addition, the word given to the network should be the vector representation. There are many methods (e.g., Word2Vec) that convert words to vectors in natural language processing [10]. Therefore, we add an embedding layer before LSTM, which embeds words into dense representations. In our experiments, we set embedding size to 300 and set hidden units in LSTM to 50.



**Fig. 2.** Overview of the LSTM network used for the title field. For example, “A softbot-based interface to the internet.” is the title in one citation. Firstly, the title is divided into word list. Then, words are converted to vectors by the embedding layer. Finally, each word vector is sequentially entered into the network. The final hidden state is provided for next step.

**Character-Based RNN.** The structure of author and venue fields is similar in one citation. They are both composed of several words and the relevance between words is weak. Thus word-based LSTM is not suitable for them, while character-based RNN may be qualified for this task. For example, in the “O. Etzioni, D. Weld.”, the word list is “O”, “Etzioni”, “D” and “Weld”. When the word “O” occurs, it’s hard to predict what is the next word, because it can be any other words in the list. However, in one word, the characters are relevant. For example, the characters “er” can be added after the word “sing” to become the word “singer”. Considering the characteristic, we split the author and venue fields into character list and retain their original order. Therefore, we adopt character-based RNN to capture their local semantic information in the proposed network. Figure 3 illustrates the RNN. Similar to word-based LSTM, the author and venue fields are divided into several characters, and then each character is processed in order. The final hidden state is used as input for next step. In addition, each character from the author and venue fields is converted to vector representation by adding embedding layer before the input layer of RNN.



**Fig. 3.** Overview of the RNN network used for the author and venue fields. For example, “R.S. Sutton” is one author’s name. We remove invalid characters (i.e., space and dot) and split the author into the character list. The following processing steps are similar with word-based LSTM.

## 4 Experiments

In this section, we conducted a set of experiments to validate the effectiveness of our method on three datasets.

### 4.1 Datasets

Some experiments were conducted on two real datasets (Citeseer and Cora [3]) and one synthetic dataset (MIX-citations). For each citation, we only utilized author, title and venue fields in our experiments. In addition, we integrated Citeseer with Cora to generate a synthetic dataset, called MIX-citations. To increase the complexity of the synthetic dataset, we adopted a noise strategy to expand the scale and complexity, including swapping two words, deleting one word and swapping or deleting characters in certain words randomly. In the experiments, we adopted precision, recall, F1 score and accuracy for measuring the effectiveness of the proposed method.

### 4.2 Comparison with Baseline Methods

In the experiments, we compared the proposed method with five baseline methods including two metric-based methods [1,7], a learning-based method [3] and two RNN-based methods [12,13]. Table 2 shows the performance of each method on three datasets. From the table, some observations can be made as follows: (1) RNN-based methods have better performance than traditional metric-based and learning-based methods. This is mainly due to the outstanding learning ability of RNN on sequence data. The recurrent architecture in RNN makes it have a strong contextual memory ability which captures local semantical information better. (2) RNN-based methods have an anti-noise ability. As we mentioned above, we employ a noise strategy to enhance the complexity and expand the scale on the synthetic dataset. However, all three RNN-based methods still have

**Table 2.** Performance comparison with baseline methods on three datasets.

Dataset	Method	Precision	Recall	F1 Score	Accuracy
Citeseer	MaLSTM [12]	93.93	88.57	91.17	90.07
	biLSTM [13]	96.11	94.28	95.19	94.48
	Block+Attribute Weighted Sum [7]	79.29	78.13	78.71	-
	Attribute+Relation Weighted Sum [1]	87.21	74.77	80.51	-
	Learned Vector Space [3]	-	-	88.00	-
	Our Method	<b>96.79</b>	<b>95.87</b>	<b>96.33</b>	<b>95.77</b>
Cora	MaLSTM [12]	98.82	97.59	98.20	97.18
	biLSTM [13]	98.92	99.45	99.18	98.71
	Block+Attribute Weighted Sum [7]	72.95	92.62	81.62	-
	Attribute+Relation Weighted Sum [1]	88.05	77.19	82.26	-
	Learned Vector Space [3]	-	-	80.30	-
	Our Method	<b>99.99</b>	<b>99.69</b>	<b>99.83</b>	<b>99.75</b>
MIX-citations	MaLSTM [12]	96.99	96.36	96.67	95.11
	biLSTM [13]	97.88	98.09	97.98	97.03
	Our Method	<b>98.97</b>	<b>98.40</b>	<b>98.68</b>	<b>98.07</b>

a good performance on MIX-citations dataset. (3) Our method outperforms the other RNN-based methods. The main reason is that the citation is not a continuous sentence and there are information gaps between its neighboring fields. Our joint multi-field siamese recurrent neural network uses different RNN networks to extract features of corresponding fields and fuse the multi-field features, which can effectively reduce the impact of information gaps. However, other RNN-based methods consider the citation as a whole sentence, which ignore information gaps between different fields in one citation.

### 4.3 Effectiveness of Different Network Components

For one citation, the title filed is a short sentence, so the local semantic between words are strong. However, since the author and venue fields consist of independent phrases, a weak relationship exists between phrases. Therefore, word-based LSTM and character-based RNN are employed for title and author/venue fields, respectively. To validate the effectiveness of our proposed method, we used different network components in the proposed framework and evaluated their performance on three datasets, which are reported in Table 3. According to experimental results, we can observe that on Citeseer dataset, our method outperforms all

**Table 3.** Performance on three datasets with different network components in the proposed framework.

Dataset	Method	Precision	Recall	F1 Score	Accuracy
Citeseer	biRNN+LSTM+biRNN	96.10	93.96	95.02	94.30
	MNN+LSTM+MNN	94.96	95.87	95.41	94.66
	LSTM+LSTM+LSTM	96.06	93.01	94.51	93.75
	RNN+RNN+RNN	90.81	87.93	89.35	87.86
	RNN+LSTM+RNN (ours)	<b>96.79</b>	<b>95.87</b>	<b>96.33</b>	<b>95.77</b>
Cora	biRNN+LSTM+biRNN	99.61	<b>99.84</b>	99.72	99.57
	MNN+LSTM+MNN	94.96	95.87	95.41	94.66
	LSTM+LSTM+LSTM	96.06	93.01	94.51	93.75
	RNN+RNN+RNN	98.53	99.22	98.88	98.22
	RNN+LSTM+RNN (ours)	<b>99.99</b>	99.69	<b>99.83</b>	<b>99.75</b>
MIX-citations	biRNN+LSTM+biRNN	98.73	<b>99.10</b>	<b>98.92</b>	<b>98.40</b>
	MNN+LSTM+MNN	98.87	98.60	98.74	98.14
	LSTM+LSTM+LSTM	<b>99.01</b>	98.54	98.78	98.20
	RNN+RNN+RNN	96.27	96.63	96.45	94.76
	RNN+LSTM+RNN (ours)	98.97	98.40	98.68	98.07

other methods which choose different combination of neural networks, including RNN, Bidirectional RNN, LSTM and Multi-layer Neural Network. In addition, on Cora and MIX-citations datasets, although the performance of our method is not always the best, the gap between the proposed method and the best one is less than 0.8%. The main reason might be that the variants of RNN have stronger anti-noise ability.

## 5 Conclusion

In this paper, we present a novel joint multi-field siamese recurrent neural network for entity resolution in citation domain. Especially, word-based LSTM and character-based RNN are adopted for the title and author/venue in one citation, respectively. The proposed method can fully exploit the contextual information of each field and fuse the multi-field features effectively. We conducted a set of experiments on three datasets. Experimental results show that the proposed method achieves a better performance than other baseline methods and other RNN-based methods. Moreover, the effectiveness of word-based LSTM and character-based RNN in the proposed method is validated by experiments.

## References

1. Bhattacharya, I., Getoor, L.: Iterative record linkage for cleaning and integration. In: Proceedings of the 9th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, pp. 11–18. ACM (2004)
2. Bhattacharya, I., Getoor, L.: Relational clustering for multi-type entity resolution. In: Proceedings of the 4th International Workshop on Multi-relational Mining, pp. 3–12. ACM (2005)
3. Bilenko, M., Mooney, R.J.: Adaptive duplicate detection using learnable string similarity measures. In: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 39–48. ACM (2003)
4. Brizan, D.G., Tansel, A.U.: A survey of entity resolution and record linkage methodologies. *Commun. IIMA* **6**(3), 5 (2006)
5. Christen, P.: Febrl-: an open source data cleaning, deduplication and record linkage system with a graphical user interface. In: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1065–1068. ACM (2008)
6. Cohen, W., Ravikumar, P., Fienberg, S.: A comparison of string metrics for matching names and records. In: KDD Workshop on Data Cleaning and Object Consolidation, vol. 3, pp. 73–78 (2003)
7. Elmagarmid, A.K., Ipeirotis, P.G., Verykios, V.S.: Duplicate record detection: a survey. *IEEE Trans. Knowl. Data Eng.* **19**(1), 1–16 (2007)
8. Getoor, L., Machanavajjhala, A.: Entity resolution: theory, practice & open challenges. *Proc. VLDB Endowment* **5**(12), 2018–2019 (2012)
9. Hadsell, R., Chopra, S., LeCun, Y.: Dimensionality reduction by learning an invariant mapping. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 1735–1742. IEEE (2006)
10. Lai, S., Liu, K., He, S., Zhao, J.: How to generate a good word embedding. *IEEE Intell. Syst.* **31**(6), 5–14 (2016)
11. Minton, S.N., Nanjo, C., Knoblock, C.A., Michalowski, M., Michelson, M.: A heterogeneous field matching method for record linkage. In: Fifth IEEE International Conference on Data Mining, 8 p. IEEE (2005)
12. Mueller, J., Thyagarajan, A.: Siamese recurrent architectures for learning sentence similarity. In: AAAI, pp. 2786–2792 (2016)
13. Neculoiu, P., Versteegh, M., Rotaru, M.: Learning text similarity with siamese recurrent networks. In: Proceedings of the 1st Workshop on Representation Learning for NLP, pp. 148–157 (2016)



# Using Machine Learning for Determining Network Robustness of Multi-Agent Systems Under Attacks

Guang Wang<sup>1</sup>, Ming Xu<sup>2</sup>, Yiming Wu<sup>2</sup>(✉), Ning Zheng<sup>1</sup>, Jian Xu<sup>1</sup>,  
and Tong Qiao<sup>2</sup>

<sup>1</sup> School of Computer Science and Technology, Hangzhou Dianzi University,  
Hangzhou 310018, China

<sup>2</sup> School of Cyberspace, Hangzhou Dianzi University, Hangzhou 310018, China  
yimgwu@hotmail.com

**Abstract.** Network robustness has been the key metric in the analysis of secure distributed consensus algorithms for multi-agent systems (MASs). However, it is proved that determining the network robustness of a MASs with large nodes is NP-hard. In this paper, we try to apply machine learning method to determine the robustness of MASs. We use neural network (NN) that consists of Multilayer Perceptions (MLPs) to learn the representation of multi-agent networks and use softmax as our classifiers. We compare our method with a traditional CNN-based approach on a graph-structured dataset. It is shown that with the help of machine learning method, determining robustness can be possible for MASs with large nodes.

**Keywords:** Network robustness · Machine learning  
Multi-agent systems

## 1 Introduction

In recent years, much attention has been devoted to the study of consensus problem for multi-agent systems (MASs) with an emphasis on cyber security. Network robustness [4], a specific property of network topology, measuring redundancy of directed edges between all pairs of nonempty, disjoint subsets of nodes in a network, which plays an important role for consensus algorithms to be able to withstand a subset of nodes failed or compromised. In previous work [4, 12], the authors adopted the network robustness as metrics for analysis of resilient consensus algorithms that use only local neighboring information. Hence, determining the robustness of a network is important for determining whether resilient consensus algorithms can be work. However, in [10, 11] the authors proved that determining the robustness of MASs is NP-hard. In a more specific work [3], the authors carried out algorithm for determining robustness of a network, and indicated that for a MAS with  $n$  nodes, its complexity is  $\mathcal{O}(n^2 3^n)$ .

Recently, machine learning has received significant attention because of its powerful ability of learning features from different kinds of data. In the field of representation learning, there are many works [5,7,9] trying to apply the convolutional neural network on the graph-structure data. However, to the best of our knowledge, there are very few machine learning works on multi-agent networks.

In this paper, we attempt to use machine learning method to determine network robustness. We firstly extract statistical features from adjacent matrices related to multi-agent networks. Then we get the particular features from spectral space by performing spectral clustering [8] on symmetric adjacent matrices that are transformed from origin matrices by a simple symmetric method [6]. After that we feed the combined features to a neural network that consists of MLPs to learn representation of networks. Finally, we use softmax as our classifiers to determine network robustness.

Our contributions are two-fold. Firstly, to the best of our knowledge, we are the first one to use machine learning method to determine network robustness. Secondly, we depict graph-structured data in statistical point of view with extra features from spectral space.

## 2 Preliminary

We firstly give two main definitions of network robustness and then introduce some useful lemmas about it briefly.

**Definition 1.** [3] (*(r, s)-edge reachable set*): Given a nontrivial digraph  $\mathcal{D}$  and a nonempty subset of nodes  $\mathcal{S}$ , we say that  $\mathcal{S}$  is an  $(r, s)$ -edge reachable set if there are at least  $s$  nodes in  $\mathcal{S}$  with at least  $r$  in-neighbors outside of  $\mathcal{S}$ , where  $r, s \in \mathbb{Z}_{\geq 0}$ ; i.e., given  $\mathcal{X}_{\mathcal{S}}^r = \{i \in \mathcal{S} : |\mathcal{N}_i^{in} \setminus \mathcal{S}| \geq r\}$ , then  $|\mathcal{X}_{\mathcal{S}}^r| \geq s$ .

**Definition 2.** [3] (*(r, s)-robustness*): A nonempty, nontrivial digraph  $\mathcal{D} = (\mathcal{V}, \mathcal{E})$  on  $n$  nodes ( $n \geq 2$ ) is  $(r, s)$ -robust, for nonnegative integers  $r \in \mathbb{Z}_{\geq 0}, 1 \leq s \leq n$ , if for every pair of nonempty, disjoint subsets  $\mathcal{S}_1$  and  $\mathcal{S}_2$  of  $\mathcal{V}$  at least one of the following holds (recall  $\mathcal{X}_{\mathcal{S}_k}^r = \{i \in \mathcal{S}_k : |\mathcal{N}_i^{in} \setminus \mathcal{S}_k| \geq r\}$  for  $k \in \{1, 2\}$ ):

- (i)  $|\mathcal{X}_{\mathcal{S}_1}^r| = |\mathcal{S}_1|$ ;
- (ii)  $|\mathcal{X}_{\mathcal{S}_2}^r| = |\mathcal{S}_2|$ ;
- (iii)  $|\mathcal{X}_{\mathcal{S}_1}^r| + |\mathcal{X}_{\mathcal{S}_2}^r| \geq s$ .

The following are some important lemmas that will help us to better understand the network robustness.

**Lemma 1.** [4] For any  $(r, s)$ -robust digraph  $\mathcal{D}$  also meets  $(r', s')$ -robust when  $0 \leq r' \leq r, 1 \leq s' \leq s$ .

**Lemma 2.** [4] For any  $(r, s)$ -robust digraph  $\mathcal{D}$ , the  $r$  and  $s$  satisfy the following conditions:

$$\begin{cases} 0 \leq r \leq \min(\delta^{in}, \lceil \frac{n}{2} \rceil), \\ 0 \leq s \leq n. \end{cases} \quad (1)$$



Where  $n$  is the number of nodes.  $\delta^{in}$  is the minimum in-degree of  $\mathcal{D}$ .

**Lemma 3.** For any digraph  $\mathcal{D}$ , if the number of node set is odd, then  $K_n$  is the only digraph on  $n$  nodes that is  $(\lceil \frac{n}{2} \rceil, s)$ -robust with  $s \geq \lfloor \frac{n}{2} \rfloor$ , where  $K_n$  is a complete digraph.

From Lemma 1, one can get that there are hierarchical relations among different  $(r, s)$ -robust networks. For example, if the multi-agent network satisfies  $(3, 4)$ -robust then it also satisfies  $(3, 3)$ -robust,  $(2, 4)$ -robust and for any  $1 \leq r \leq 3$  and  $1 \leq s \leq 4$ . Lemma 2 gives us the upper bound of  $r$  and  $s$ . Lemma 3 implies that the training data will be unbalanced.

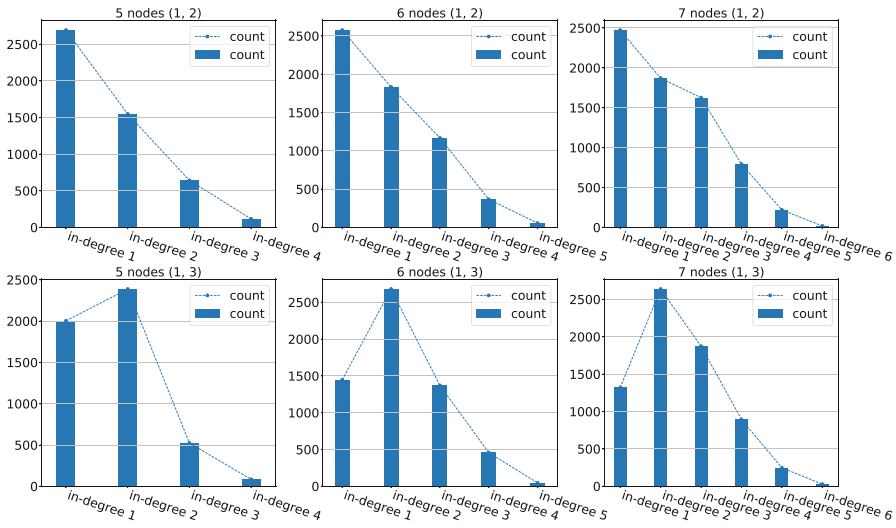


Fig. 1. Bar diagram of counts of in-degree which sizes range from 5 to 7.

### 3 The Method

#### 3.1 Preprocessing Steps

Firstly, the graphs are generated by Erdős-Rényi random graph model [1]. We change  $p$  randomly to generate different graphs. Then we calculate  $(r, s)$ -robust of each graph by deterministic algorithm and make it as a label of a graph. To better understand the labeled graph-structured data, we study the distributions of in-degree for different  $(r, s)$ -robust networks. One example of distributions of in-degree of some networks are showed in Fig. 1. As we can see, in each column, the shape of distribution of in-degree is changed when the number of  $(r, s)$ -robust increased. However, in each row, networks that own different sizes of node set but

**Table 1.** Table of Pearson correlation coefficients

Correlation to $r$	Correlation to $s$	Descriptions
0.9/0.8	0.1/0.1	The minimum of in/out-degrees
0.1/0.1	-0.1/-0.1	The count of minimum of in/out-degrees
0.7/0.7	0.2/0.3	The maximum of in/out-degrees
0.4/0.4	0.0/0.1	The count of maximum of in/out-degrees
0.9/0.9	0.2/0.2	The mean of in/out-degrees
-0.5/-0.4	0.0/0.1	The variance of in/out-degrees
0.8/0.8	0.3/0.2	The mean of mode of in/out-degrees
0.8/0.8	0.2/0.2	The mean of median of in/out-degrees

the same robustness, have the similar in-degree distribution. Just because of this we extract some statistics from the distributions of in-degree of networks and calculate Pearson correlation coefficients between the statistics and  $(r, s)$ -robust. The results are illustrated in Table 1.

### 3.2 Spectral Clustering Steps

In order to extend our features, we execute spectral clustering on networks to get particular features from spectral space. However, the targets of spectral clustering are usually undirected networks. In order to solve this problem, we just use symmetric method [6] to get the symmetric adjacent matrix  $\mathbf{U}$ . The formula is as follows:

$$\mathbf{U} = \mathbf{A} + \mathbf{A}^T \quad (2)$$

Then we employ spectral clustering on the symmetric adjacent matrix. It can be formulated as follows:

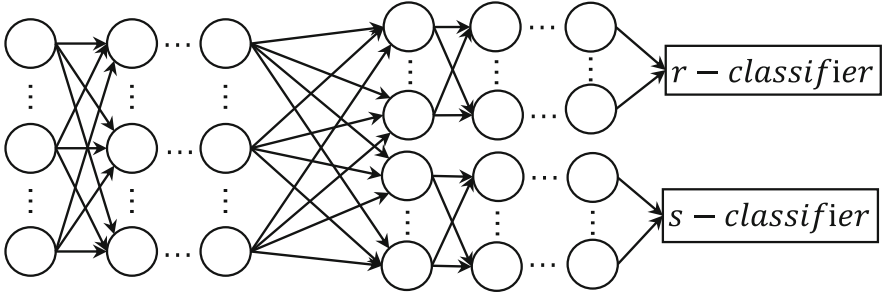
$$\min_{\mathbf{F}} \sum_{i,j=1}^N u_{ij} \|\mathbf{f}_i - \mathbf{f}_j\|_2^2 = \text{tr}(\mathbf{F}^T \mathbf{L} \mathbf{F}), \quad \text{s.t. } \mathbf{F}^T \mathbf{F} = \mathbf{I} \quad (3)$$

where  $\text{tr}(\cdot)$  denotes the trace function,  $\mathbf{f}$  denotes the eigenvector,  $\mathbf{L}$  is the Laplacian matrix [8] obtained from  $\mathbf{U}$ , and  $\mathbf{I}$  denotes the identity matrix.

Then we employ  $k$ -means algorithm on eigenvectors from all symmetric adjacent matrices. We will have one integer feature for each cluster identified through the  $k$ -means algorithm. The value of each graph feature is determined by analyzing the eigenvectors related to a network. The  $k^{\text{th}}$  feature will indicate the number of eigenvectors that have been assigned to the cluster  $C_k$ .

### 3.3 Network Robustness Classifier Details

In this section, the combined features are fed into the Neural Network (NN) model for feature learning. In general, NNs consist of Multilayer Perceptions



**Fig. 2.** Diagram of Neural Network Model.

(MLPs) which contain one or more hidden layers with multiple hidden units or neurons. The mathematical expression of a NN with  $L$  hidden layers can be defined as follows:

$$f(\mathbf{x}) = \sigma(\mathbf{W}^L \dots \sigma(\mathbf{W}^2 \sigma(\mathbf{W}^1 \mathbf{x} + \mathbf{b}^1) + \mathbf{b}^2) \dots + \mathbf{b}^L) \quad (4)$$

where  $\mathbf{x}$  is a vector of input features,  $\mathbf{W}$  is a matrix of weights,  $\mathbf{b}$  is a vector of biases,  $\sigma$  is an activation function.

After passing through a number of neurons, the input features are fed into classifiers. The classification loss can be represented as follows:

$$C(f(\mathbf{x}), y) = \ell(f(\mathbf{x}), y) \quad (5)$$

where  $y$  corresponds to the ground truth label, which indicates the  $(r, s)$ -robust of a network, and  $\ell$  indicates the classification function.

The diagram of our neural network structure is illustrated in Fig. 2. The value of  $r$  and  $s$  of network robustness not only has its own loss function and classifier but also shares with a number of parameters.

## 4 Experiment

### 4.1 Datasets and Compared Methods

Here we set the probability  $p$  of dataset from 0.1 to 0.9, then randomly generates different  $(r, s)$ -robust networks whose sizes range from 5 to 10. To evaluate the performance of our method, we compare it with two other methods, namely CNN-GN and O-NN-SOFT. All of the three methods are summarized as follows:

*CNN-GN*: It is a graph-structured data learning method [9]. It consists of graph reordering, structural augmentation, and CNN.

*O-NN-SOFT*: origin adjacent matrix related to a multi-agent network are fed into the NN for supervised learning and then use softmax as classifiers. This method is treated as a baseline.

*F-NN-SOFT*: Our method proposed in this paper.

## 4.2 Experiment Setting

For CNN-GN, we firstly perform the simple symmetric method on all our synthetic data. Then, we execute spectral clustering on every networks to finish graph reordering step. We set  $K$  of  $k$ -means from 3 to 5 and augment weights of all the outliers, the edge outside clusters, by  $\epsilon$  which is larger than 1. After augmenting step, all the new networks are joining together as channels. Finally we construct a convolution neural network which consists of three convolution layers, followed by three fully connected layers of which two branches are four fully connected layers respectively.

For O-NN-SOFT and F-NN-SOFT, the network structures are same which consists of five fully connected layers of which two branches are four fully connected layers respectively. But for F-NN-SOFT, we firstly execute simple symmetric method on our synthetic data and then perform spectral clustering on each symmetric matrix. We set  $K$  of  $k$ -means the number of node set. This is because we want to our F-NN-SOFT can capture different patterns in light of  $s$  whose upper bound is equal to the number of nodes.

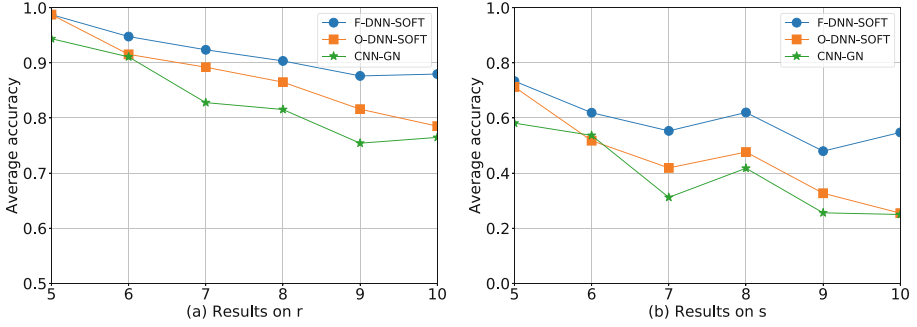
All experiments used 10-fold cross validation. We used Gaussian distribution to initialize weights and biases, where the distribution is  $\mathcal{N}(0, 0.1)$ . We set batch size of 1024, and for regularization, a dropout rate of 0.5 and early stopping with 150 epochs. The softmax-cross-entropy was optimized with Adam [2] with a initial learning rate of 0.001.

## 4.3 Experiment Results

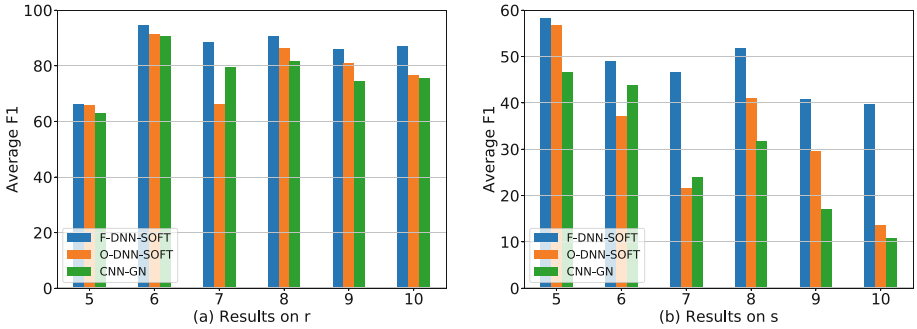
In this section, we evaluate our method for determining network robustness. The average classification performance are shown in Table 2. We use accuracy

**Table 2.** Performances of the compared methods

Dataset	Measures	F-DNN-SOFT		O-DNN-SOFT		CNN-GN	
		$r$	$s$	$r$	$s$	$r$	$s$
5	AVG ACC	98.75	73.37	98.75	71.25	94.38	58.13
	AVG F1	66.33	58.28	65.83	56.78	62.89	46.61
6	AVG ACC	94.77	61.92	91.54	51.77	91.08	53.69
	AVG F1	94.40	49.04	91.31	37.19	90.61	43.83
7	AVG ACC	92.37	55.32	89.21	41.89	82.79	31.21
	AVG F1	88.42	46.67	66.12	21.59	79.44	23.87
8	AVG ACC	90.35	62.00	86.50	47.65	81.54	41.77
	AVG F1	90.46	51.78	86.39	41.09	81.65	31.71
9	AVG ACC	87.63	47.97	81.60	32.70	75.43	25.60
	AVG F1	85.80	40.82	80.87	29.59	74.43	17.11
10	AVG ACC	87.97	54.77	78.52	25.48	76.48	25.00
	AVG F1	86.93	39.61	76.77	13.68	75.48	10.86



**Fig. 3.** Average accuracy of  $r$  and  $s$  on synthetic graph-structured data.



**Fig. 4.** Average F1 of  $r$  and  $s$  on synthetic graph-structured data.

and F1 score as our evaluation metrics. As we can see that our method F-DNN-SOFT outperforms all the other methods on the synthetic data. Compared with baseline algorithm O-DNN-SOFT that using adjacent matrix as input, our method achieves better performance. In addition we can see with the increase of size of node set, our method is more robust and the performance decreases slowly as showed in Figs. 3 and 4. O-DNN-SOFT may not capture the features influenced by  $s$  so it has very poor accuracy and F1 score in determining  $s$ . CNN-GN is the worst one among all the methods. This may because CNN-GN can not discriminate how many nodes inside the cluster have neighbors outside the cluster. For example, if here exists 4 edges outside the cluster, these edges may belong to only one node inside the cluster or may belong two nodes that one has 1 edge and the other one has 3 edges, or may be another combinations.

## 5 Conclusion

In this paper, we have made a try to determine robustness of MASs by machine learning. We compared our method with two traditional CNN-based methods on synthetic graph-structured data which generated by Erdős-Rényi random

graph model. The results showed that our method has a good performance on determining network robustness.

**Acknowledgment.** This work was supported by the cyberspace security Major Program in National Key Research and Development Plan of China under grant 2016YFB0800201, Natural Science Foundation of China under grants 61572165 and 61702150, State Key Program of Zhejiang Province Natural Science Foundation of China under grant LZ15F020003, Key Research and Development Plan Project of Zhejiang Province under grants 2017C01062 and 2017C01065, Public Research Project of Zhejiang Province under grant LGG18F020015, and Scientific Research fund of Zhejiang Provincial Education Department under grant Y201737924.

## References

1. Erdős, P., Rényi, A.: On random graphs, I. *Publicationes Mathematicae (Debrecen)* **6**, 290–297 (1959)
2. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. arXiv preprint [arXiv:1412.6980](https://arxiv.org/abs/1412.6980) (2014)
3. LeBlanc, H.J., Koutsoukos, X.D.: Algorithms for determining network robustness. In: *Proceedings of the 2nd ACM international conference on High confidence networked systems*, pp. 57–64. ACM (2013)
4. LeBlanc, H.J., Zhang, H., Koutsoukos, X., Sundaram, S.: Resilient asymptotic consensus in robust networks. *IEEE J. Sel. Areas Commun.* **31**(4), 766–781 (2013)
5. Niepert, M., Ahmed, M., Kutzkov, K.: Learning convolutional neural networks for graphs. In: *International Conference on Machine Learning*, pp. 2014–2023 (2016)
6. Satuluri, V., Parthasarathy, S.: Symmetrizations for clustering directed graphs. In: *Proceedings of the 14th International Conference on Extending Database Technology*, pp. 343–354. ACM (2011)
7. Tixier, A.J.P., Nikolentzos, G., Meladianos, P., Vazirgiannis, M.: Classifying graphs as images with convolutional neural networks. arXiv preprint [arXiv:1708.02218](https://arxiv.org/abs/1708.02218) (2017)
8. Von Luxburg, U.: A tutorial on spectral clustering. *Stat. Comput.* **17**(4), 395–416 (2007)
9. Wang, S., He, L., Cao, B., Lu, C.T., Yu, P.S., Ragin, A.B.: Structural deep brain network mining. In: *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 475–484. ACM (2017)
10. Zhang, H., Fata, E., Sundaram, S.: A notion of robustness in complex networks. *IEEE Trans. Control Netw. Syst.* **2**(3), 310–320 (2015)
11. Zhang, H., Sundaram, S.: Robustness of complex networks with implications for consensus and contagion. In: *2012 IEEE 51st Annual Conference on Decision and Control (CDC)*, pp. 3426–3432. IEEE (2012)
12. Zhang, H., Sundaram, S.: Robustness of information diffusion algorithms to locally bounded adversaries. In: *2012 American Control Conference (ACC)*, pp. 5855–5861. IEEE (2012)



# Collective Hyper-heuristics for Self-assembling Robot Behaviours

Shuang Yu<sup>1</sup>, Andy Song<sup>2</sup>, and Aldeida Aleti<sup>1</sup>

<sup>1</sup> Monash University, Clayton 3168, Australia  
shuang.yu@monash.edu

<sup>2</sup> RMIT University, Melbourne 3000, Australia

**Abstract.** Swarm robots are highly desirable in dealing with complex tasks. However, manual coding of individual robot behaviours and robot collaboration is not trivial especially under unknown and dynamic environments. This study introduced a hyper-heuristic methodology for this challenge, so robots can learn suitable behaviours during the process. The hyper-heuristic method creates actions based on a set of low-level heuristics and improves these actions through autonomous heuristic adjustment. A collective negotiation and updating mechanism is proposed so the robot swarm performance can be improved. We evaluate this method on the problem of building surface cleaning. Experiments show the effectiveness of the hyper-heuristic method and the collective learning mechanism.

**Keywords:** Hyper heuristics · Swarm robots · Collective behaviours

## 1 Introduction

Collectively, robots are able to complete more complex and larger scale tasks than a single robot. However, it is not a trivial task to coordinate and control a collection of robots. Instead of manually synthesising sophisticated control strategies for complex tasks, we introduce an online learning based hyper-heuristic approach for robot swarms. With the hyper-heuristic approach, only low level operators are supplied to generate solutions instead of full control strategies. These operators are elements for constructing instructions for various tasks and environments. With online learning, these instructions can be adapted to deal with changes in the environment.

Hyper-heuristic methods have been used on complex problems, such as bin-packing, timetabling and vehicle routing. They aim to provide high quality solutions across a wide variety of problem domains, rather than developing tailor-made methodologies for each problem [6]. Hyper-heuristics search for solvers instead of solutions. Hyper-heuristics can be decentralised and involve multiple agents. [4] proposed hyper-heuristics that enable several heuristic selection strategies to happen in parallel among multiple virtual agents, and the agents collectively accept or reject selected heuristics through group decision making strategies.

For swarm robots, group behaviours emerge from local control laws. [3] defined the problem of swarm behaviour composition, and proposed an offline learning method to automatically generate behaviour sequences for a human operator to execute. The algorithm operates in a known and static environment with obstacles, and lacks the ability to cope with unknown or dynamic environments. Additionally, the existing methods are tailored for their intended applications, meaning that manual re-design of the algorithms are required from task to task. Hyper-heuristics, on the other hand, have been used to automatically generate algorithms for new problems.

To evaluate our method, a case study on self-assembling robots is used, where the task is to clean multiple surfaces. Basic moves and operators are supplied for the hyper-heuristic engine to build cleaning strategies. For different surface layouts, different combinations of swarm behaviours need to be used. Robots need to be able to autonomously connect to cross gaps, and scatter to navigate and clean.

## 2 Methodology

In our early study a hyper-heuristic framework is proposed [7]. The proposed improvements are presented below.

Robots iteratively learn its own heuristic score table according to the applied heuristics' past performances. In each time interval, after updating the heuristic score table for the robot swarm, each individual robot may have a different score. Hence a collective decision making strategy is proposed to determine the best heuristic for the whole swarm for the next time interval. [7] investigated voting for decision making, which requires each robot to vote on a candidate heuristic. In this paper we propose two strategies to identify the heuristic collectively. For a group of  $N$  robots and  $K$  heuristics, the heuristic score matrix of the swarm is:

$$\hat{H} = \begin{bmatrix} \hat{h}^{11} & \hat{h}^{12} & \hat{h}^{13} & \dots & \hat{h}^{1K} \\ \hat{h}^{21} & \hat{h}^{22} & \hat{h}^{23} & \dots & \hat{h}^{2K} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \hat{h}^{N1} & \hat{h}^{N2} & \hat{h}^{N3} & \dots & \hat{h}^{NK} \end{bmatrix}$$

The first strategy calculates the sum of each heuristic score across all robots in the group, and the heuristic with maximum total score becomes the chosen one:

$$k_{selected} = \arg \max_{1 \leq k \leq K} \sum_{n=1}^N \hat{h}^{nk} \quad (1)$$

This strategy aims to maximise the total group performance score for the next iteration, so is named as MAX-SUM.



The second strategy is named as MAX-MIN as it finds the heuristic with the best minimum score. It the minimum score of each heuristic, and selects the heuristic that has the maximum minimum score.

$$k_{selected} = \arg \max_{1 \leq k \leq K} \{ \arg \min_{1 \leq n \leq N} \hat{h}^{nk} \} \quad (2)$$

In software optimisation problems, after a selected heuristic is applied to the problem, hyper-heuristic uses a move acceptance method to determine if the heuristic, or the “move” should be accepted or rejected [1]. In a robotic system, rejecting a heuristic means going back to the robot’s original state (position), which in the real-world is impractical and consumes energy. Therefore in our framework, the group always accepts the selected group heuristic. Robots will then update their heuristic to be used for the next iteration.

### 3 Self-assembling Swarm Robot Cleaner

For self-assembling robots, swarm behaviours emerge from the physical connections and interactions between robots. Self-assembling robots can connect to cross gaps, scatter to cover surfaces, or flock to stay close but separated. A repository of such basic behaviours  $B = \{b^1, b^2 \dots b^k\}$ , defines the set of robot control laws that read sensor data, and execute actions accordingly. The task is then, *given a set of self-assembling robot behaviours, and an objective function, construct a sequence of such behaviours autonomously to maximize the objective value  $f(t)$  in unknown environments.*

#### 3.1 Implementation

To solve this problem, we consider a type of swarm robot behaviour as a *heuristic*, which is defined by the control rules that take in environmental input and control actions periodically, the robots collectively evaluate, select and update the next behaviours, such as to assemble or to scatter.

The objective function measures the performance of robot behaviours on the given objective, and each robot is programmed with local controllers to carry out actions corresponding to the heuristics in the heuristics repository. Robots start with an initial heuristic, and the same score  $\hat{h}_0^i$  for each heuristic in the repository, which is based on the assumption that the environment is unknown. The robots perform actions for a period of time guided by the initial heuristic, and learn heuristic scores online according to the method described in [7]. Any robot can request heuristic scores from other robots to determine the next heuristic. In order to physically achieve the collective decision process described in Sect. 2, the robots send communication messages to each other following the diagrams in Fig. 1.

### 3.2 Heuristic Repository

The heuristic repository for this cleaning task contains: sweeping, bridging, exploring, circling and flocking, as described below:

All the heuristics have sufficient collision avoidance and gap avoidance mechanisms programmed into them, therefore robot self-preservation is not an issue.

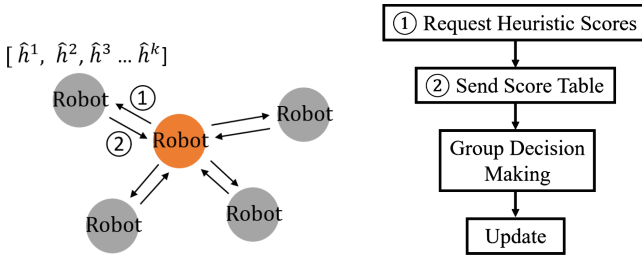


Fig. 1. Implementation of group decision making.

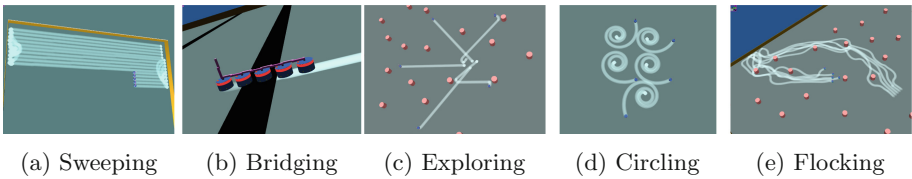


Fig. 2. Heuristic repository

As shown in Fig. 2, **sweeping** behaviour allows robots to physically connect to form a vertical line, and perform horizontal back-and-forth movements. **Bridging** behaviour enables the swarm to connect and form a bridge structure. It allows a group of robots to cross gaps and small obstacles, as a connection is formed between robots to keep them from falling when they lose contact with the surface. **Exploring** behaviour allows the robots to scatter in random directions (which change at random time intervals) on the surface. **Circling** performs a spiral motion, following with straight lines in random directions. This cleaning pattern is widely used by cleaning robots. The **flocking** local controller enables robots to follow a flocking behaviour [5], as illustrated in Fig. 2. This heuristic allows robots to stay relatively close to each other, making it easier for the swarm to assemble when needed, while preserving some degree of random exploration.

## 4 Experiments and Results

The experiments are set up in Webots Simulator with emulated real-world physics [2]. Given the repository of behaviours in Sect. 3.2, robots in our experiments could physically connect to move onto a different surface, scatter to move past obstacles, or explore uncleaned areas. Since no single behaviour is able to clean multiple surfaces, a combination of heuristics is a must for this scenario. Five robots are used in simulations, and the robot model is based on the non-holonomic robots in [7], as shown in Fig. 2.

They are differential wheeled robots with a dirt sensor, obstacle sensors, wireless communication, suction cups, cleaning wipes and a gripper arm to connect to a neighbouring robot. The robots are simulated to clean at the maximum speed of  $2.98 \text{ m}^2$  per time interval of 40s, which we define as one unit area, and collect 14.9% of the available dirt at each pass.

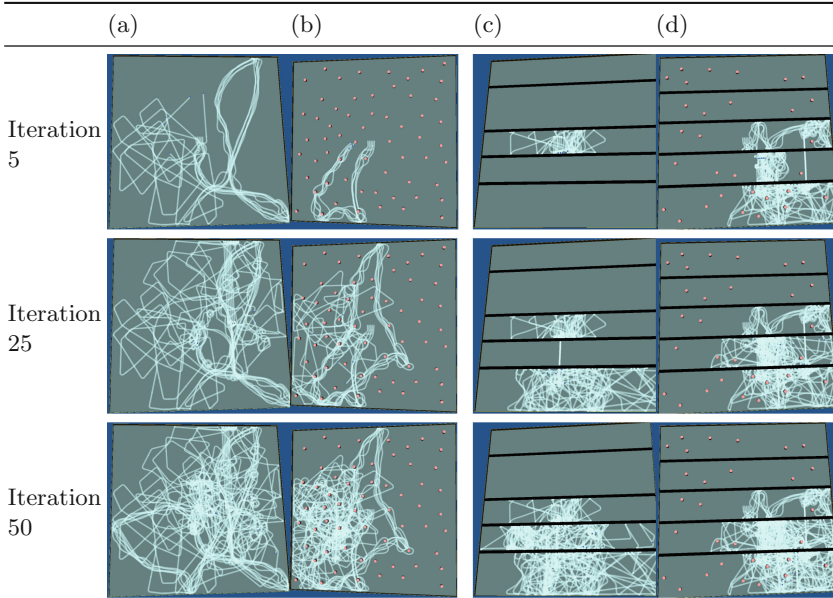
The objective here is to clean surfaces efficiently. Therefore the objective function can be defined as  $f(t) = \frac{\sum(L_t D_t)}{T}$  where  $L_t$  is the distance travelled since the last measure,  $D_t$  is the dirtiness reading from the dirt sensor at iteration  $t$ . Iteration length  $T$  represents the period that the swarm applies one heuristic before re-selecting. The termination criterion is set to be terminating after 50 iterations, which is adequate to show the characteristics of the constructed behaviour sequences.

### 4.1 Robustness in Different Environments

The hyper-heuristic implementation is tested in four types of environments shown in Fig. 3. Environment (a) is a flat surface that is  $8 \text{ m} \times 8 \text{ m}$ , bounded by four barriers; environment (b) is the same size, with 50 obstacles; (c) has four gaps on the surface, which single robots cannot cross; (d) has four gaps and 30 obstacles. The obstacles and gaps are randomly placed in the environment, and are different in each experimental run. Robots have no prior knowledge of which heuristics are suitable for a specific task.

Results from Scenarios (a) and (b) show that the robots are able to perform the cleaning task continuously and robustly with the presence of walls and obstacles, and those from (c) and (d) show that robots are able to move across surfaces to clean.

To show the effectiveness of the collective decision making, we compared the results with the benchmark method in [7] where iterative voting is used to determine the best heuristic over 30 runs under each experimental condition. Table 1 detailed the performance increase with MAX-SUM hyper-heuristics and MAX-MIN hyper-heuristics. Mann-Whitney U tests have been performed on the results. An ‡ symbol indicates a p value  $< 0.05$  comparing to voting, while ⊖ indicates a p value  $> 0.05$ .



**Fig. 3.** Cleaning progress of the swarm at 5, 25 and 50 iterations in four types of environmental layouts: (a) flat empty surface, (b) surface with obstacles (indicated by red blobs), (c) five surfaces separated by gaps (black stripes), and (d) five separated surfaces with obstacles.

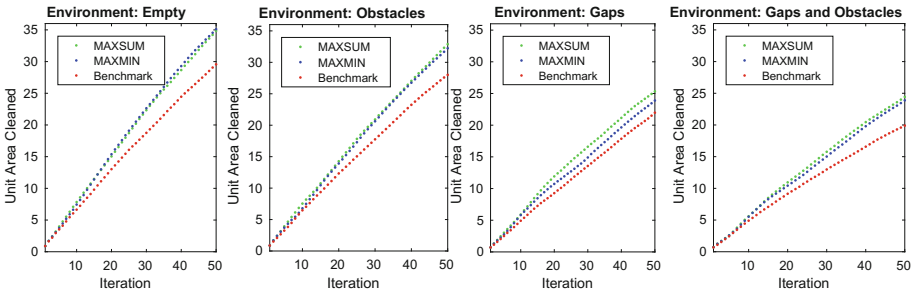
**Table 1.** Comparing Mean and Std. MAX-SUM, MAX-MIN with voting in surface cleaning over 30 runs

Environment	Strategy	Iteration 5	Iteration 10	Iteration 25	Iteration 50	Std.
Empty	Voting	3.5190	6.6389	16.0767	29.5652	2.6209
	MAX-SUM	3.9862 ‡	7.8382 ‡	18.7146 ‡	34.8393 ‡	2.0073
	MAX-MIN	3.7504 ⊖	7.4416 ‡	19.0949 ‡	35.1158 ‡	1.8367
Obstacles	Voting	3.2409	6.4697	15.0475	28.0052	3.1566
	MAX-SUM	3.9173 ‡	7.5398 ‡	17.8297 ‡	32.8875 ‡	2.3756
	MAX-MIN	3.7029 ‡	6.7881 ⊖	17.2879 ‡	32.2359 ‡	2.0985
Gaps	Voting	2.5055	4.9182	11.4324	21.9716	3.2797
	MAX-SUM	3.0322 ‡	5.9019 ‡	14.3134 ‡	25.3342 ‡	2.4588
	MAX-MIN	2.7628 ⊖	5.8101 ‡	12.7129 ‡	23.8899 ‡	2.1364
Gaps and Obstacles	Voting	2.4350	4.8998	11.0722	19.8920	2.8743
	MAX-SUM	2.5812 ⊖	5.6010 ‡	13.2607 ‡	24.4293 ‡	3.6064
	MAX-MIN	2.5870 ⊖	5.5249 ‡	12.6025 ‡	23.8831 ‡	1.9721
Overall	Voting	2.9251	5.7317	13.4072	24.8585	5.0173
	MAX-SUM	3.3792 ‡	6.7203 ‡	16.0296 ‡	29.3726 ‡	5.2839
	MAX-MIN	3.2008 ‡	6.3912 ‡	15.4245 ‡	28.7812 ‡	5.4000

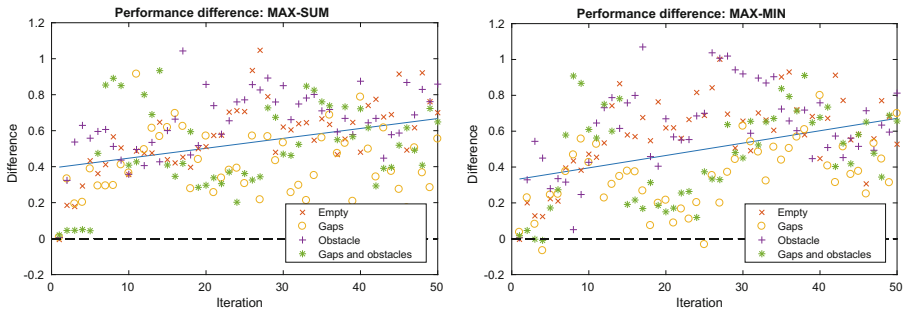
### 4.2 Comparison of Collective Decision Making Strategies

In this part we compare the two proposed group decision making strategies MAX-SUM and MAX-MIN with a benchmark method (majority vote) in [7]. The benchmark method requires a candidate heuristic to be selected and proposed to the group, and according to each robot’s acceptance criteria, the heuristic accepted by the majority of the swarm will be the group heuristic.

Each group performs cleaning tasks for 30 runs each under the four types of layouts. Figure 4 plots the mean performance of the proposed group decision making strategies compared to benchmark for every iteration. Both MAX-SUM and MAX-MIN outperform benchmark in all four environments, overall by 18% and 15.8% respectively. After applying Mann-Whitney U tests on MAX-SUM performances and MAX-MIN overall performances over 30 runs, we have found that under the environment with gaps, the p value is  $0.0096 < 0.05$ , indicating that MAX-SUM is better than MAX-MIN. For the three other environments there are no statistically differences between these two approaches.



**Fig. 4.** Comparing mean performance of MAX-SUM, MAX-MIN and the benchmark method over 30 experimental runs



**Fig. 5.** No-learning vs. MAX-SUM and MAX-MIN on four scenarios.

In Fig. 5, we plot the performance improvements by our methods compared with no learning during the task execution process. No learning means randomly selecting heuristics at all decision points. The Y-axis of Fig. 5 is the performance difference at each iteration calculated from

$$\frac{\sum_{i=1}^t (f_i^L - f_i^{NL})}{\sum_{i=1}^t f_i^{NL}}$$

where  $f_i^L$  is the total area been cleaned at iteration  $t$  by the swarms with learning, and  $f_i^{NL}$  is without learning. It can be seen that for the majority (97%) of iterations, the differences are greater than 0. Furthermore, the differences increase over time, as indicated by the blue line which represents the trend. By the end of 50 iterations, both MAX-SUM and MAX-MIN have achieved approximately 67% improvement over the no-learning counterparts.

**Table 2.** Comparing the time to reach consensus in the collective decision making

Num. of robots	MAX-SUM & MAX-MIN	Voting
5	0.256 s	4 s
10	0.256 s	32 s
20	0.256 s	65 s

As the scale of the swarm increases, the time costed performing online learning depends heavily on the collective decision making strategy, as the initial individual learning is independent of other robots. We compare the time to reach consensus on during the collective decision making process. Table 2 shows the comparison of robot swarms with size 5, 10 and 20 robots respectively. As can be seen, using MAX-SUM and MAX-MIN, the decision time is much faster and does not increase with the number of robots. In contrast, the time used for voting is longer and increases quite rapidly with the number of robots. That means the complexity of voting is higher than that of MAX-SUM and MAX-MIN. For large scale swarm robots, MAX-SUM and MAX-MIN would be advantageous.

## 5 Conclusions and Future Works

This paper presented a novel decentralised hyper-heuristics learning method for robot swarms and proposed collective decision making mechanisms. Experiments on these methods have been conducted on swarm cleaning robots. The results show that the proposed hyper-heuristics achieved significant improvement in task performance through the collective decision making. Further experiments show this mechanism is effective and faster. Hence we conclude that online hyper-heuristic learning is effective for swarm robots, as is the proposed collective decision making mechanism.

In this study, all robots in a group apply the same heuristic during the same time interval, and we will further investigate heterogeneous swarms where robots in the group apply different heuristics.

## References

1. Burke, E., Kendall, G., Newall, J., Hart, E., Ross, P., Schulenburg, S.: Hyper-heuristics: an emerging direction in modern search technology. In: Glover, F., Kochenberger, G.A. (eds.) *Handbook of Metaheuristics*, pp. 457–474. Springer, Boston (2003). [https://doi.org/10.1007/0-306-48056-5\\_16](https://doi.org/10.1007/0-306-48056-5_16)
2. Michel, O.: Webots: symbiosis between virtual and real mobile robots. In: Heudin, J.-C. (ed.) *VW 1998. LNCS (LNAI)*, vol. 1434, pp. 254–263. Springer, Heidelberg (1998). [https://doi.org/10.1007/3-540-68686-X\\_24](https://doi.org/10.1007/3-540-68686-X_24)
3. Nagavalli, S., Chakraborty, N., Sycara, K.: Automated sequencing of swarm behaviors for supervisory control of robotic swarms. In: *IEEE International Conference on Robotics and Automation*, pp. 2674–2681. IEEE (2017)
4. Özcan, E., Misir, M., Kheiri, A.: Group decision making hyper-heuristics for function optimisation. In: *UK Workshop on Computational Intelligence*, pp. 327–333. IEEE (2013)
5. Reynolds, C.W.: Flocks, herds and schools: a distributed behavioral model. *ACM SIGGRAPH Comput. Graph.* **21**(4), 25–34 (1987)
6. Sabar, N.R., Ayob, M., Kendall, G., Qu, R.: A dynamic multiarmed bandit-gene expression programming hyper-heuristic for combinatorial optimization problems. *IEEE Trans. Cybern.* **45**(2), 217–228 (2015)
7. Yu, S., Aleti, A., Barca, J., Song, A.: Hyper-heuristic online learning for self-assembling swarm robots. In: *International Conference on Computer Science*. Springer, Heidelberg (2018, to appear)



# Matrix Factorization for Identifying Noisy Labels of Multi-label Instances

Xia Chen<sup>1</sup>, Guoxian Yu<sup>1</sup>, Carlotta Domeniconi<sup>2</sup>, Jun Wang<sup>1</sup>,  
and Zili Zhang<sup>1</sup>

<sup>1</sup> College of Computer and Information Sciences,  
Southwest University, Chongqing 400715, China  
xchen@email.swu.edu.cn, {gxyu, kingjun, zhangz1}@swu.edu.cn

<sup>2</sup> Department of Computer Science, George Mason University,  
Fairfax, VA 22030, USA  
carlotta@cs.gmu.edu

**Abstract.** Current effort on multi-label learning generally assumes that the given labels are noise-free. However, obtaining noise-free labels is quite difficult and often impractical. In this paper, we study how to identify a subset of relevant labels from a set of candidate ones given as annotations to instances, and introduce a matrix factorization based method called *MF-INL*. It first decomposes the original instance-label association matrix into two low-rank matrices using nonnegative matrix factorization with feature-based and label-based constraints to retain the geometric structure of instances and label correlations. MF-INL then reconstructs the association matrix using the product of the decomposed matrices, and identifies associations with the lowest confidence as noisy associations. An empirical study on real-world multi-label datasets with injected noisy labels shows that MF-INL can identify noisy labels more accurately than other related solutions and is robust to input parameters. We empirically demonstrate that both feature-based and label-based constraints contribute to boosting the performance of MF-INL.

**Keywords:** Multi-label learning · Noisy labels identification  
Low-rank matrix factorization

## 1 Introduction

Multi-label classification models the scenario where each instance is associated with a set of labels, and its goal is to find a set of relevant labels for unlabeled instances [6, 36]. Multi-label classification has attracted ever-increasing interest in the context of text classification [18], automatic image annotation [23], and protein function prediction [28], among other applications. Currently, multi-label learning methods mainly focus on how to assign a set of appropriate labels to unlabeled instances [35], how to replenish missing labels for incompletely labeled instances [19], and how to make use of interrelationships of labels [33]. Most of



the aforementioned methods assume that the assigned labels are correct. However, things may go awry in practice, and the collected label set of observed instances may include *noisy* (or not applicable) labels. This is because the labels of multi-label instances are collected by human annotators with wide-ranging levels of expertise and different techniques [14, 29].

Despite the progress achieved in multi-label learning, the problem of identifying noisy labels in multi-label instances, to the best of our knowledge, is *seldom* studied. Its goal is the selection of a set of appropriate labels for a multi-label instance, by removing from the given collection those that do not apply. As such, the problem is more challenging and different from partial-label learning [4, 34], which identifies one label from a set of candidate labels of an instance, disregarding their interrelationship.

In this paper, we propose a matrix factorization based approach called MF-INL to identify noisy labels of multi-label instances. MF-INL first factorizes the instance-label association matrix into two low-rank matrices via graph regularized Nonnegative Matrix Factorization (NMF) [2, 13]. Particularly, MF-INL takes advantage of the geometric structure among instances and correlations between labels to define two graphs, and thus enforces the factorized matrices to be consistent with both the geometric structure and label correlations. After that, MF-INL reconstructs the association matrix using the product of the two factorized matrices, and considers the reconstructed associations with low entry values as noisy labels of instances. Experimental results on publicly available multi-label datasets show that MF-INL can identify noisy labels of multi-label instances more accurately than other related methods [3, 15, 27, 31, 34].

## 2 Related Work

Partial-label learning studies the scenario in which an instance is associated with a set of candidate labels among which only one is valid [4, 26], and can be viewed as a special case of multi-label partial-label learning, where we force each instance to be annotated with one label only. Most partial-label learning methods combine the ground-truth label identification and classifier training on the over-labeled instances [32]. Some methods treat equally all candidate labels and make prediction by averaging the outputs of all candidate labels [4, 8]. Other methods assume a parametric model and consider the ground-truth labels as latent variables, which are iteratively refined to disambiguate candidate labels [16, 21]. More recent methods follow two stages: they first directly disambiguate the candidate labels, and then perform classification on the disambiguated labels of instances [31, 34]. All the aforementioned partial-label learning methods assume that each instance is associated with exactly one ground-truth label. But in many data mining application domains, instances are naturally associated with multiple labels.

LSDR methods have been proposed for multi-label learning to handle large label space [20]. CPLST [3] simultaneously considers both the instance-label association information and the instance-feature information to minimize the upper bound of popular Hamming loss, and consequently to seek a latent label space. FaIE [15] jointly maximizes the recoverability of the original label space from the latent space, and the predictability of the latent space from the feature space, to simultaneously learn the coding and decoding matrices, which are used to compress and recover the label space, respectively. All these LSDR based methods aim to find an optimal low-dimensional subspace with respect to the original label space, and to perform multi-label learning in the subspace to improve performance by removing irrelevant, redundant, or noisy information (i.e., noisy labels of instances) [3]. In addition, some other recent methods also show the contribution of label embedding for multi-label classification [24, 25].

Motivated by the robustness to noise of low-rank matrix approximation [7, 17], we introduce a matrix factorization based approach MF-INL to identify noisy labels of multi-label instances. The empirical study shows that MF-INL can identify noisy labels more accurately than competitive algorithms.

### 3 Proposed Method

Let  $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n] \in \mathbb{R}^{d \times n}$  be  $n$  multi-label instances in the  $d$ -dimensional feature space.  $\mathbf{Y} \in \mathbb{R}^{q \times n}$  is the known instance-label association matrix.  $\mathbf{Y}_{ci} \in \{0, 1\}$ , where  $\mathbf{Y}_{ci} = 1$  if  $\mathbf{x}_i$  is associated with the  $c$ -th label,  $\mathbf{Y}_{ci} = 0$  otherwise. The goal of MF-INL is to identify noisy associations in  $\mathbf{Y}$ .

The low-rank approximation of a *noisy* matrix is robust to noise [12, 17]. Thus, in this paper, we consider to seek the low-rank approximation of the original instance-label association matrix to identify noisy labels. Particularly, we advocate to decompose the instance-label association matrix  $\mathbf{Y}$  into two low-rank matrices by NMF, which is a widely used low-rank matrix decomposition method. Then, we can use the product of two low-rank matrices to approximate the original matrix. However, this decomposition does not consider the geometric structure among instances and correlations among labels, both of which should be leveraged to guide the decomposition. Our proposed method MF-INL addresses this issue by integrating nonnegative matrix factorization with feature-based and label-based constraints. MF-INL minimizes the objective function as follows:

$$\psi(\mathbf{U}, \mathbf{V}) = \|\mathbf{Y} - \mathbf{UV}^T\|^2 + \alpha \text{tr}(\mathbf{V}^T \mathbf{L}^F \mathbf{V}) + \beta \text{tr}(\mathbf{U}^T \mathbf{L}^L \mathbf{U}) \quad (1)$$

$\mathbf{U} \in \mathbb{R}^{q \times r} \geq 0$  and  $\mathbf{V} \in \mathbb{R}^{n \times r} \geq 0$  are the low-dimensional representations of the original  $\mathbf{Y}$  matrix based on rows and columns, respectively ( $r \ll q, r \ll n$ ).  $\alpha$  and  $\beta$  are the positive scalar parameters. By integrating two regularizations into NMF, MF-INL enforces the factorized low-rank matrices to preserve the geometric structure of instances and the interrelationships of labels in a coherent and coordinated manner. We will elaborate on the two regularization constraints in the following subsections.

### 3.1 Feature-Based Regularization

Features of an instance essentially decide its outputs (labels) [11, 30]. To leverage the geometric structure of instances, which depends on the instance features, we adopt the manifold assumption, which is widely-used in dimensionality reduction and semi-supervised learning [1, 22]. The manifold assumption assumes that if two data points  $\mathbf{x}_i$  and  $\mathbf{x}_j$  are close in the intrinsic geometry of the ambient space, they should have similar outputs (or labels). Firstly, we construct a  $k$  nearest neighbor graph [9, 10] to model the local geometric structure of  $n$  instances as follows:

$$\mathbf{W}_{ij}^F = \begin{cases} 1, & \text{if } \mathbf{x}_j \in \mathcal{N}_k(\mathbf{x}_i) \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

where  $\mathcal{N}_k(\mathbf{x}_i)$  is the set of  $k$  nearest neighbors of  $\mathbf{x}_i$ , and the neighborhood relationship is determined using the Euclidean distance. Here, the 0–1 weighting scheme is used to weighting the edges of the  $k$ NN graph, but other weighting schemes and distance metrics can also be defined based on the specific application domains.  $\mathbf{v}_i$  is the low-dimensional representation of  $i$ -th column of  $\mathbf{Y}$ . The manifold assumption to enforce that  $\mathbf{v}_i$  and  $\mathbf{v}_j$  are close to each other when  $\mathbf{x}_i$  and  $\mathbf{x}_j$  are close in the original ambient space is specified as follows:

$$\psi_1(\mathbf{V}) = \frac{1}{2} \sum_{i,j=1}^n \|\mathbf{v}_i - \mathbf{v}_j\|^2 \mathbf{W}_{ij}^F = \text{tr}(\mathbf{V}^T \mathbf{L}^F \mathbf{V}) \quad (3)$$

where  $\mathbf{D}^F$  is a diagonal matrix with  $\mathbf{D}_{ii}^F = \sum_{j=1}^n \mathbf{W}_{ij}^F$ ,  $\mathbf{L}^F = (\mathbf{D}^F - \mathbf{W}^F)$  is the graph Laplacian matrix, and  $\text{tr}(\cdot)$  denotes the trace of a matrix. By minimizing  $\psi_1(\mathbf{V})$ , the proximity between close instances in the original space can be preserved in the low-dimensional space spanned by  $\mathbf{V}$  through the feature-based regularization.

### 3.2 Label-Based Regularization

In multi-label learning, labels are not mutually exclusive, and different pairs of labels (or groups of labels) may have different degrees of correlation. In contrast, traditional multi-class classification and partial-label learning explicitly (or implicitly) assumes the labels are mutually exclusive. Various types of label correlations have been explored in multi-label learning and they generally can improve the performance [33]. Given this, besides feature-aware constraint, we also define a label-aware constraint.

Each row of  $\mathbf{U}$  can be viewed as a low-dimensional (or latent) label vector of the original label vector expressed by the corresponding row of  $\mathbf{Y}$ . The expectation is that the new low-dimensional representation is able to preserve the interrelationship of labels in the original space, that is, if labels  $s$  and  $t$  are correlated in  $\mathbf{Y}$ , then the low-dimensional representations of  $\mathbf{Y}_s$  and  $\mathbf{Y}_t$ ,

i.e.  $\mathbf{u}_s$  and  $\mathbf{u}_t$ , would also be correlated in the latent label space. We first measure the label correlation using the cosine similarity as follows:

$$\mathbf{W}_{st}^L = \frac{\mathbf{Y}_s \mathbf{Y}_t^T}{\|\mathbf{Y}_s\| \|\mathbf{Y}_t\|} \quad (4)$$

where  $\mathbf{Y}_s$  is the  $s$ -th row of  $\mathbf{Y}$ , and  $\mathbf{W}_{st}^L \in [0, 1]$  denotes the label correlation between the  $s$ -th and  $t$ -th labels.  $\mathbf{W}_{st}^L$  is large when  $s$  and  $t$  frequently co-occur as an annotation of instances, and is small otherwise. We use cosine similarity for its simplicity, but other measures can be also used. To preserve label correlations in the latent label space, we defines the label-aware constraint for  $\mathbf{U}$  as follows:

$$\psi_2(\mathbf{U}) = \frac{1}{2} \sum_{s,t=1}^q \|\mathbf{u}_s - \mathbf{u}_t\|^2 \mathbf{W}_{st}^L = \text{tr}(\mathbf{U}^T \mathbf{L}^L \mathbf{U}) \quad (5)$$

where  $\mathbf{D}^L$  is a diagonal matrix with  $\mathbf{D}_{st}^L = \sum_{t=1}^q \mathbf{W}_{st}^L$  and  $\mathbf{L}^L = \mathbf{D}^L - \mathbf{W}^L$ . As in Eq. (3), by minimizing Eq. (5), we can preserve correlations in the latent label space.

As for the standard NMF, we optimize Eq. (1) using an iterative algorithm. Readers can refer to [2] for details. By minimizing Eq. (1), MF-INL reconstructs the approximated association matrix as  $\hat{\mathbf{Y}} = \mathbf{U}\mathbf{V}^T$ . After the reconstruction, the associations available in  $\mathbf{Y}$ , but inconsistent with the low-rank representation with respect to the geometric structure of instances and to the correlations between labels, have low values in  $\hat{\mathbf{Y}}$ ; otherwise have high values. As a result, each entry of  $\hat{\mathbf{Y}}$  reflects the association confidence between a particular instance and a particular label. The labels corresponding to the smaller entries in  $\hat{\mathbf{Y}}$  are more likely to be deemed as noisy labels.

## 4 Experimental Setup

**Datasets:** To study the performance of MF-INL, we conduct experiments on four multi-label datasets (listed in Table 1, downloaded from the Mulan Library<sup>1</sup>). Since there are no off-the-shelf multi-label datasets that can be directly used to validate the performance of identifying noisy labels of multi-label instances, we assume that the available labels of instances in these four datasets are noise-free, and randomly inject additional  $p \times q$  labels to each instance as noisy labels, where  $p$  is the ratio of noisy labels. Specially, to study the performance of MF-INL under different levels of noise, we conduct experiments with  $p$  set to 0.3 and 0.5.

**Comparative Methods:** We compare MF-INL against IPAL [31], PL-LEAF [34], CPLST [3], FaIE [15], ProDM [27]. These methods have been presented in the Sect. 2.

<sup>1</sup> <http://mulan.sourceforge.net/datasets-mlc.html>.

**Table 1.** Datasets used in the experiments. Avg-Labels is the average number of labels per instance.

Dataset	Instances	Features	Labels	Avg-Labels
Enron	1702	1001	53	3.378
Yeast	2417	103	14	4.237
Rcv1-s5	6000	47235	101	2.642
Tmc	28596	500	22	2.220

**Evaluation Metrics:** We use three representative multi-label learning and partial-label learning evaluation metrics: RankingLoss (RL), OneError (OE), and AveragePrecision (AP) [36]. They can evaluate the identification of noisy labels from the perspective of label distribution [5]. Note that the smaller the values of RL and OE, the better the performance is; while the larger the values of AP, the better the performance is. We report the  $1-RL$  and the  $1-OE$  in the following experiments. As such, *larger* values imply a *better* performance.

## 5 Experimental Results and Analysis

### 5.1 Noisy Label Identification

Following the experimental protocol in partial-label learning [34], we considered all instances in each dataset as both training and testing data. We search the optimal parameter values for  $\alpha$  and  $\beta$  in the range  $\{0, 0.01, 0.1, 1, 10, 100, 1000, 10000\}$ . As a result,  $\alpha$  and  $\beta$  are set to 0.1 and 100, respectively. The neighborhood size  $k$  is set to 5. The same value  $r = 10$  is used for MF-INL, CPLST, and FaIE. Other parameters of comparing methods are set (or optimized) as suggested by the authors in their code, or respective papers. Table 2 reports the average results of 10 independent runs of all methods under each particular  $p$ . From Table 2, MF-INL outperforms the other methods across all the metrics in most cases. This observation shows that MF-INL can identify noisy labels of multi-label instances more accurately than other related methods, and supports our motivation to factorize the instance-label association matrix into latent low-rank subspaces.

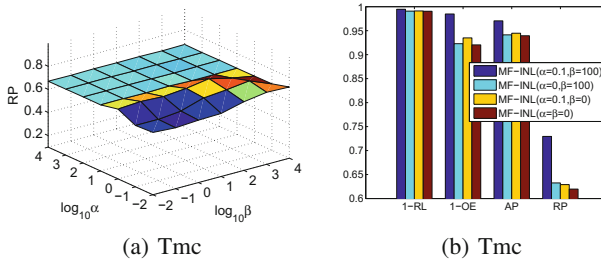
To study the efficiency of MF-INL, we record the runtime of all other comparing methods on a server with configuration: CentOS 7, 256 GB RAM and Intel Exon E5-2678v3. The total runtime (seconds) of CPLST, FaIE, ProDM, IPAL, PL-LEAF and MF-INL on all datasets is 173, 208, 306, 2933, 1962475 and 108, respectively. From these observations and the results in Table 2, we can conclude that MF-INL not only holds comparable runtime against efficient counterparts, but also achieves a superior performance.

**Table 2.** Performance for the identification of noisy labels as the ratio ( $p$ ) of randomly injected noisy labels per instance increases. ●/○ indicates whether MF-INL is statistically (according to a pairwise  $t$ -test at 95% significance level) superior/inferior to the other method for a particular value of  $p$ .

	$p$	CPLST	FaIE	ProDM	IPAL	PL-LEAF	MF-INL
Enron	1-RL	0.3 0.799 ± 0.006●	0.776 ± 0.003●	0.996 ± 0.000●	0.992 ± 0.000●	0.994 ± 0.000●	0.998 ± 0.000
		0.5 0.812 ± 0.007●	0.784 ± 0.004●	0.995 ± 0.000●	0.987 ± 0.000●	0.990 ± 0.000●	0.996 ± 0.000
	1-OE	0.3 0.973 ± 0.004●	0.971 ± 0.003●	0.964 ± 0.003●	0.922 ± 0.004●	0.982 ± 0.003●	0.985 ± 0.004
		0.5 0.963 ± 0.004●	0.960 ± 0.003●	0.936 ± 0.004●	0.868 ± 0.005●	0.966 ± 0.000●	0.970 ± 0.005
	AP	0.3 0.801 ± 0.006●	0.780 ± 0.003●	0.951 ± 0.002●	0.910 ± 0.002●	0.943 ± 0.000●	0.973 ± 0.002
		0.5 0.804 ± 0.007●	0.779 ± 0.004●	0.930 ± 0.002●	0.853 ± 0.002●	0.909 ± 0.001●	0.953 ± 0.002
Yeast	1-RL	0.3 0.970 ± 0.004●	0.975 ± 0.001●	0.977 ± 0.001●	0.960 ± 0.001●	0.966 ± 0.000●	0.985 ± 0.001
		0.5 0.936 ± 0.006●	0.932 ± 0.003●	0.956 ± 0.001●	0.930 ± 0.001●	0.943 ± 0.001●	0.970 ± 0.001
	1-OE	0.3 0.971 ± 0.006○	0.976 ± 0.003○	0.941 ± 0.005●	0.888 ± 0.004●	0.917 ± 0.002●	0.967 ± 0.003
		0.5 0.885 ± 0.016●	0.885 ± 0.011●	0.914 ± 0.004●	0.831 ± 0.003●	0.893 ± 0.001●	0.945 ± 0.004
	AP	0.3 0.955 ± 0.005●	0.960 ± 0.003●	0.952 ± 0.002●	0.914 ± 0.002●	0.930 ± 0.001●	0.969 ± 0.001
		0.5 0.892 ± 0.008●	0.884 ± 0.008●	0.919 ± 0.002●	0.867 ± 0.002●	0.896 ± 0.001●	0.946 ± 0.003
Tmc	1-RL	0.3 0.903 ± 0.004●	0.879 ± 0.001●	0.995 ± 0.000	0.989 ± 0.000●	0.992 ± 0.000●	0.995 ± 0.000
		0.5 0.903 ± 0.002●	0.879 ± 0.002●	0.991 ± 0.000○	0.980 ± 0.000●	0.991 ± 0.000	0.990 ± 0.001
	1-OE	0.3 0.929 ± 0.005●	0.937 ± 0.002●	0.968 ± 0.001●	0.869 ± 0.001●	0.993 ± 0.000○	0.987 ± 0.001
		0.5 0.885 ± 0.003●	0.894 ± 0.005●	0.917 ± 0.001●	0.765 ± 0.001●	0.979 ± 0.000○	0.939 ± 0.005
	AP	0.3 0.886 ± 0.004●	0.872 ± 0.001●	0.971 ± 0.000	0.929 ± 0.000●	0.955 ± 0.000●	0.971 ± 0.002
		0.5 0.860 ± 0.002●	0.849 ± 0.003●	0.940 ± 0.001○	0.868 ± 0.001○	0.959 ± 0.000○	0.937 ± 0.004
Rcv1-s5	1-RL	0.3 0.584 ± 0.003●	0.573 ± 0.002●	0.998 ± 0.000●	0.998 ± 0.000●	0.996 ± 0.000●	0.999 ± 0.000
		0.5 0.591 ± 0.004●	0.577 ± 0.002●	0.997 ± 0.000●	0.996 ± 0.000●	0.993 ± 0.000●	0.998 ± 0.000
	1-OE	0.3 0.881 ± 0.001●	0.881 ± 0.001●	0.923 ± 0.004●	0.943 ± 0.002●	0.907 ± 0.000●	0.992 ± 0.002
		0.5 0.878 ± 0.001●	0.876 ± 0.001●	0.875 ± 0.004●	0.886 ± 0.003●	0.868 ± 0.000●	0.978 ± 0.004
	AP	0.3 0.589 ± 0.003●	0.578 ± 0.002●	0.948 ± 0.002●	0.945 ± 0.001●	0.911 ± 0.000●	0.979 ± 0.001
		0.5 0.592 ± 0.004●	0.577 ± 0.002●	0.915 ± 0.002●	0.902 ± 0.002●	0.872 ± 0.000●	0.963 ± 0.002

## 5.2 Parameter Sensitivity Analysis

To investigate the sensitivity of  $\alpha$  and  $\beta$ , we vary  $\alpha$  and  $\beta$  in the range  $\{0.01, 0.1, 1, 10, 100, 1000, 10000\}$  with  $p = 0.3$ , and report the average RP of MF-INL in 10 independent runs under different combinations of  $\alpha$  and  $\beta$  in Fig. 1(a). From this figure, we can see that MF-INL achieves a stable and good performance for a wide range of  $\alpha$  and  $\beta$  values. In addition, we can see that MF-INL, with values  $\alpha = 0.01$  and  $\beta = 0.01$ , has lower RP than many other values' combinations. This observation suggests that it's necessary to integrate feature-based and label-based regularizations into NMF to obtain coherent matrices. To further investigate the influence of feature-based and label-based regularizations, we test the performance of MF-INL under extreme settings of  $\alpha$  and  $\beta$ , that is:  $\alpha = 0$  and  $\beta = 100$ ;  $\alpha = 0.1$  and  $\beta = 0$ ; and we report the results of MF-INL under these extreme settings in Fig. 1(b). From the results we can see that using feature-based and label-based regularizations together can significantly improve the performance of NMF. Using either one of the two regularizations gives comparable or better performance than NMF alone.  $\alpha = 0$  and  $\beta = 0$ .



**Fig. 1.** *RP* of MF-INL under different combinations of  $\alpha$  and  $\beta$  on Tmc.

The rank size of the decomposed (projected) matrix is an essential parameter for MF-INL and LSDR-based methods CPLST and FaIE. We also conduct experiments to study the sensitivity of  $r$ . Due to page limitation, we do not provide the results in this paper. From the results, MF-INL is robust to different input values of  $r$ , while CPLST and FaIE are sensitive to  $r$ . Besides, MF-INL outperforms CPLST and FaIE under each considered value of  $r$ . The robustness of MF-INL to  $r$  can be attributed to the fact that MF-INL can find coherent low-rank matrices by simultaneously preserving the geometric structure of instances and the correlation of labels.

## 6 Conclusions and Future Work

In this paper, we study an interesting but rarely explored problem of multi-label learning: identifying noisy labels of multi-label instances. To solve this problem, we introduce a matrix factorization based method called MF-INL.

The experimental study shows that MF-INL can identify noisy labels more accurately than other competitive techniques. It will be interesting to study the performance of MF-INL under different choices of distance metrics and label correlations, and to iteratively update the correlations, since label correlations are affected by noise in the original instance-label association matrix.

**Acknowledgments.** This work is supported by Natural Science Foundation of China (61741217 and 61402378), Natural Science Foundation of CQ CSTC (cstc2016jcyjA0351), Open Research Project of Hubei Key Laboratory of Intelligent Geo-Information Processing (KLGIP-2017A05) and Chongqing Graduate Student Research Innovation Project [No. CYS18089].

## References

1. Belkin, M., Niyogi, P., Sindhwani, V.: Manifold regularization: a geometric framework for learning from labeled and unlabeled examples. *JMLR* **7**(11), 2399–2434 (2006)
2. Cai, D., He, X., Han, J., Huang, T.S.: Graph regularized nonnegative matrix factorization for data representation. *TPAMI* **33**(8), 1548–1560 (2011)
3. Chen, Y., Lin, H.: Feature-aware label space dimension reduction for multi-label classification. In: *NIPS*, pp. 1529–1537 (2012)
4. Cour, T., Sapp, B., Taskar, B.: Learning from partial labels. *JMLR* **12**(5), 1501–1536 (2011)
5. Geng, X.: Label distribution learning. *TKDE* **28**(7), 1734–1748 (2016)
6. Gibaja, E., Ventura, S.: A tutorial on multilabel learning. *ACM Comput. Surv.* **47**(3), 52 (2015)
7. Hansen, P.C., Jensen, S.H.: FIR filter representations of reduced-rank noise reduction. *IEEE Trans. Signal Process.* **46**(6), 1737–1741 (1998)
8. Hüllermeier, E., Beringer, J.: Learning from ambiguously labeled examples. *Intell. Data Anal.* **10**(5), 419–439 (2006)
9. Jiang, L., Wang, D., Cai, Z., Jiang, S., Yan, X.: Scaling up the accuracy of k-nearest-neighbour classifiers: a Naïve-Bayes hybrid. *Int. J. Comput. Appl.* **31**(1), 36–43 (2009)
10. Jiang, L., Cai, Z., Wang, D., Zhang, H.: Bayesian Citation-KNN with distance weighting. *Int. J. Mach. Learn. Cybern.* **5**(2), 193–199 (2014)
11. Jiang, L., Zhang, L., Li, C., Wu, J.: A correlation-based feature weighting filter for Naïve Bayes. In: *TKDE* (2018). <https://doi.org/10.1109/TKDE.2018.2836440>
12. Konstantinides, K., Natarajan, B., Yovanof, G.S.: Noise estimation and filtering using block-based singular value decomposition. *IEEE Trans. Image Process.* **6**(3), 479–483 (1997)
13. Lee, D.D., Seung, H.S.: Algorithms for non-negative matrix factorization. In: *NIPS*, pp. 556–562 (2001)
14. Li, C., Sheng, V.S., Jiang, L., Li, H.: Noise filtering to improve data and model quality for crowdsourcing. *Knowl. Based Syst.* **107**, 96–103 (2016)
15. Lin, Z., Ding, G., Hu, M., Wang, J.: Multi-label classification via feature-aware implicit label space encoding. In: *ICML*, pp. 325–333 (2014)
16. Liu, L., Dietterich, T.G.: A conditional multinomial mixture model for superset label learning. In: *NIPS*, pp. 548–556 (2012)



17. Meng, D., De La Torre, F.: Robust matrix factorization with unknown noise. In: ICCV, pp. 1337–1344 (2013)
18. Nam, J., Kim, J., Mencia, E.L., Gurevych, I., Fürnkranz, J.: Large-scale multi-label text classification revisiting neural networks. In: ECML, pp. 437–452 (2014)
19. Sun, Y., Zhang, Y., Zhou, Z.: Multi-label learning with weak label. In: AAAI, pp. 593–598 (2010)
20. Tai, F., Lin, H.: Multilabel classification with principal label space transformation. *Neural Comput.* **24**(9), 2508–2542 (2012)
21. Tang, C., Zhang, M.: Confidence-rated discriminative partial label learning. In: AAAI, pp. 2611–2617 (2017)
22. Van Der Maaten, L., Postma, E., Van den Herik, J.: Dimensionality reduction: a comparative review. *JMLR* **10**, 66–71 (2009)
23. Wu, B., Lyu, S., Hu, B.G., Ji, Q.: Multi-label learning with missing labels for image annotation and facial action unit recognition. *Pattern Recogn.* **48**(7), 2279–2289 (2015)
24. Xu, C., Tao, D., Xu, C.: Robust extreme multi-label learning. In: KDD, pp. 1275–1284 (2016)
25. Yeh, C., Wu, W., Ko, W., Wang, Y.F.: Learning deep latent space for multi-label classification. In: AAAI, pp. 2838–2844 (2017)
26. Yu, F., Zhang, M.L.: Maximum margin partial label learning. *Mach. Learn.* **104**(4), 573–593 (2017)
27. Yu, G., Domeniconi, C., Rangwala, H., Zhang, G.: Protein function prediction using dependence maximization. In: ECML/PKDD, pp. 574–589 (2013)
28. Yu, G., Zhang, G., Rangwala, H., Domeniconi, C., Yu, Z.: Protein function prediction using weak-label learning. In: ACM Conference on Bioinformatics, Computational Biology and Biomedicine, pp. 202–209 (2012)
29. Zhang, J., Wu, X., Sheng, V.S.: Learning from crowdsourced labeled data: a survey. *Artif. Intell. Rev.* **46**(4), 543–576 (2016)
30. Zhang, L., Jiang, L., Li, C.: A new feature selection approach to Naive Bayes text classifiers. *Int. J. Pattern Recogn. Artif. Intell.* **30**(02), 1650003 (2016)
31. Zhang, M., Yu, F.: Solving the partial label learning problem: an instance-based approach. In: IJCAI, pp. 4048–4054 (2015)
32. Zhang, M., Yu, F., Tang, C.: Disambiguation-free partial label learning. *TKDE* **29**(10), 2155–2167 (2017)
33. Zhang, M., Zhang, K.: Multi-label learning by exploiting label dependency. In: KDD, pp. 999–1008 (2010)
34. Zhang, M., Zhou, B., Liu, X.: Partial label learning via feature-aware disambiguation. In: KDD, pp. 1335–1344 (2016)
35. Zhang, M., Zhou, Z.: ML-KNN: a lazy learning approach to multi-label learning. *Pattern Recogn.* **40**(7), 2038–2048 (2007)
36. Zhang, M., Zhou, Z.: A review on multi-label learning algorithms. *TKDE* **26**(8), 1819–1837 (2014)

## Author Index

- Ain, Qurrat Ul I-732  
Aleti, Aldeida II-499  
Alhijailan, Hajar II-100  
Al-Sahaf, Harith I-732
- Bai, Ling II-140  
Bai, Quan II-412  
Bailey, James II-192  
Bain, Michael II-300  
Bao, FeiLong I-217  
Bao, Zhijie II-132  
Barati, Molood II-412  
Bei, Xiaohui I-361  
Bhatnagar, Shobhit I-629  
Bhattacharyya, Pushpak I-629  
Bian, Naizheng II-121  
Bin, Chenzhong I-463, II-290
- Cai, Zhihua II-165  
Cao, Xiang I-797  
Cao, Yi II-29  
Chang, Liang I-463, II-290  
Chen, Fei I-1018  
Chen, Fuhai I-416  
Chen, Hao II-369  
Chen, Junyu I-954  
Chen, Lei II-265  
Chen, Qingqing I-617  
Chen, Shengyong I-759, II-309, II-360  
Chen, Siyu I-388  
Chen, Wu I-1044, II-174  
Chen, Xia II-508  
Chen, Xiang II-395  
Chen, Xingguo II-265  
Chen, Xuejiao II-219  
Chen, Yen-Wei I-617  
Chen, Zhe II-464  
Chen, Zhengpeng II-219  
Coenen, Frans I-29, II-100  
Cui, Ying I-759  
Cuiping, Li II-91
- Deng, Wu I-441  
Ding, Fuguang II-360
- Ding, Xuemei II-29  
Ding, Zhiming II-19  
Dinh, Duy-Tai I-697  
Dittakan, Kwankamon II-82  
Domeniconi, Carlotta II-508  
Dong, Hang I-29  
Dong, Jian-Feng I-126  
Dong, Rong I-1018, II-473  
Du, Yingpeng I-915, II-455  
Duan, Hua I-941  
Dukes-McEwan, Jo II-100  
Dy, Aakov II-73
- Ekbal, Asif I-629  
Elimu, Michael II-121
- Fan, Jicong II-10  
Feng, Wei I-57, II-201  
Feng, Yi II-326  
Firdaus, Mauajama I-629  
Fu, Jiamei I-684  
Fu, Mingsheng I-441  
Fu, Qiang II-29  
Fujita, Katsuhide I-786, II-404
- Gan, Yuan-Zhu I-335, I-477  
Gao, Chao II-219  
Gao, Guanglai I-217  
Gao, Jing I-604  
Gao, Shasha II-149  
Gao, Xu II-19  
Gao, Yang II-482  
Gao, Yongqiang I-559  
Ge, Hongwei I-85, I-191  
Ge, Jianjun II-1  
Ge, Jidong I-204, I-851, II-326, II-430  
Geng, Xin I-671  
Gong, Xiaolong I-402  
Gu, Haiqian II-56  
Gu, Tianlong I-463, II-290  
Guan, He II-247  
Guo, Dongyan I-759, II-360  
Guo, Lanying II-395

- Guo, Limin II-19  
 Guo, Qing I-57  
 Guo, Xiaowei I-559  
 Guo, Zichao I-531  
 Gupta, Sunil I-42, II-256
- Hadfi, Rafik I-1031  
 Hakim Newton, M. A. I-296  
 Han, Jizhong I-518  
 Han, Wei I-643  
 Han, Xianhua I-617  
 Han, Zhongxing I-604  
 Hao, Jianye II-132, II-421  
 He, Cheng II-395  
 He, Xiangdong I-980  
 Heap, Bradford II-300  
 Hong, Chen II-91  
 Hou, Chenping I-71, I-98  
 Hou, Yuexian I-658, II-282  
 Hu, Fangchao II-140  
 Hu, Haiyang I-851  
 Hu, Hao I-477  
 Hu, Hongjie I-617  
 Hu, Jing I-163  
 Hu, Qinghua I-163  
 Hu, Songlin I-518  
 Hu, Xiaohui I-604, I-837  
 Hu, Yating I-878  
 Huang, Guangyan II-19  
 Huang, Jiashuang I-1069  
 Huang, Lei I-268  
 Huang, Linpeng I-402  
 Huang, Liwei I-441  
 Huang, Longtao I-518  
 Huang, Xuhui I-559  
 Huang, Yongchuang II-29  
 Huang, Zhaowei I-463  
 Huo, Jing II-482  
 Huynh, Van-Nam I-697
- Imaeda, Teruyoshi I-1031  
 Ito, Takayuki I-1031  
 Ito, Takeshi I-256  
 Iwasa, Kosui I-786
- Ji, Genlin I-16  
 Ji, Houye I-348  
 Ji, Rongrong I-137, I-416  
 Jiang, Fei II-352
- Jiang, Liangxiao II-165, II-228  
 Jiang, Taijiao I-531  
 Jiang, Weiliang II-317  
 Jiang, Xuemeng I-772  
 Jiang, Yuan I-232, II-64  
 Jin, Hai I-150  
 Jin, Zongze I-150  
 Ju, Zhuoya II-113
- Kannangara, Sandeepa II-300  
 Katz, Gavin II-300  
 Kertkeidkachorn, Natthawut I-429  
 Kim, Kyoung-Sook I-429  
 Kim, Yusin II-73  
 Kimura, Masahiro I-282  
 Kittler, Josef II-464  
 Kong, Li I-204, I-851, II-326  
 Kowsar, Yousef II-192  
 Kuang, Haili I-463  
 Kulik, Lars II-192  
 Kurii, Mio II-404
- Lan, Long I-559  
 Leblay, Julien I-429  
 Leckie, Christopher I-891  
 Li, Bo I-1018, II-473  
 Li, Cheng II-256  
 Li, Chuanyi I-204, I-851, II-326, II-430  
 Li, Guopeng I-643  
 Li, Haichang I-604  
 Li, Hang I-1  
 Li, Jingfei I-968  
 Li, Lei I-710  
 Li, Na II-395  
 Li, Qian I-1  
 Li, Shao-Yuan I-232  
 Li, Sheng I-71  
 Li, Xianghua II-219  
 Li, Xiaohong II-132  
 Li, Xiaolin I-980  
 Li, Yinguo II-140  
 Li, Yuchen I-85  
 Li, Zechao I-375  
 Li, Zhifan I-824  
 Li, Zhixin II-38  
 Li, Zhongjin II-430  
 Li, Zhoujun II-369  
 Liang, Dong I-617  
 Liang, Hongru I-1

- Liao, Danping I-388  
 Liao, Lejian I-824, II-378  
 Lin, Jiahao I-321  
 Lin, Lan II-38  
 Lin, Lanfen I-617  
 Lin, Ruihao II-309  
 Liu, Fulai I-246  
 Liu, Gangdu I-1005  
 Liu, Guangcan I-71  
 Liu, Hong I-531  
 Liu, Hongzhi I-915, II-455  
 Liu, Jiamou I-1044, II-174  
 Liu, Junxiu II-29  
 Liu, Lu-Fei I-126, I-335  
 Liu, Qing II-412  
 Liu, Rui I-217  
 Liu, Ruyu II-309  
 Liu, Shiyu I-588  
 Liu, Shuhui I-177  
 Liu, Tong I-941  
 Liu, Wanshu II-132  
 Liu, Xin I-429  
 Liu, Yuhan I-545  
 Liu, Yunan I-928  
 Liu, Zongyue I-416  
 Lu, Jiaxin II-387  
 Lu, Jinyu II-282  
 Lu, Yao II-438  
 Luo, Bin I-204, I-851, II-326, II-430  
 Luo, Heng I-710  
 Luo, Ningqi I-746  
 Luo, Yuling II-29  
 Luo, Zhigang I-559  
 Luong, Tho Chi I-864  
 Lv, Yang II-482  
 Iwamoto, Yutaro I-617  
 Lynden, Steven I-429  
  
 Ma, Xiaotian II-10  
 Ma, Yong II-387  
 Mao, Bingcheng I-1069  
 Mao, Xiao-Jiao I-126  
 Masada, Tomonari II-156  
 Meng, Zhaopeng II-421  
 Mirmomeni, Mahtab II-192  
 Mohammed, Rafiq Ahmed II-237  
 Motoda, Hiroshi I-282  
 Mu, Weimin I-150  
 Murata, Tsuyoshi I-429  
  
 Nguyen, Thanh-Phu I-697  
 Ni, Haomiao I-531  
 Ni, Weijian I-941  
 Ning, Hao I-710  
 Niu, Zhendong I-878  
 Niu, Zhong-Han I-126  
  
 Ohara, Kouzou I-282  
 Ohsawa, Yukio I-904  
 Ong, Ethel II-73  
 Otsuka, Takano I-1031  
  
 Pan, Zhiyin II-309  
 Pang, Yuanfeng I-256  
 Peng, Chao II-395  
 Polash, M. M. A. I-296  
 Pu, Jianyu II-265  
  
 Qi, Lei II-482  
 Qian, Dongjun I-490  
 Qian, Yueliang I-531  
 Qian, Yuntao I-388  
 Qiao, Tong II-491  
 Qin, Jie I-71  
 Qiu, Chen II-165  
 Qu, Hong I-441  
 Qu, Yuanhang I-915, II-455  
  
 Rajasegarar, Sutharshan I-891  
 Ramachandran, Anil I-42  
 Ramos, Michael Joshua II-73  
 Rana, Santu I-42, II-256  
 Rashidi, Lida I-891  
 Ren, Fenghui I-113, II-343  
 Ren, Jiankang I-85, I-191  
 Ren, Yazhou I-837  
 Riahi, Vahid I-296  
 Ruan, Jianhua I-772  
 Ruan, Zhiwei I-137  
  
 Saito, Kazumi I-282  
 Sattar, Abdul I-296  
 Shang, Xuequn I-177  
 Shao, Benchu II-352  
 Shao, Ming I-310  
 Shao, Zhanpeng II-360  
 She, Dongyu I-684  
 Shen, Bin II-447  
 Shen, Chen I-416

- Shen, Ruimin II-352  
 Shi, Chuan I-348  
 Shi, Ke I-837  
 Shiratuddin, Mohd Fairuz II-237  
 Song, Andy II-499  
 Song, Binheng I-746, II-447  
 Song, Dandan I-824, II-378  
 Song, Dawei I-968, II-274  
 Song, Ge II-438  
 Su, Fei II-56  
 Su, Jinsong I-137, I-416  
 Su, Xing II-19  
 Su, Yi I-658, I-968  
 Su, Zhan I-658  
 Sun, Lei I-463, II-290  
 Sun, Liang I-191  
 Sun, Shiliang I-545, I-954  
 Sun, Wenping II-290  
 Sun, Wenxiu I-746  
 Sun, Yanpeng I-463, II-290  
 Sun, Yue I-980  
 Sun, Zhe I-1  
 Suyun, Zhao II-91
- Takasu, Atsuhiko II-156  
 Tan, Chao I-16  
 Tan, Taizhe II-47  
 Tang, Baige I-719  
 Tang, Jinhui I-375  
 Tang, Jun II-121  
 Tang, Ke I-490  
 Tang, Yanni I-1044, II-174  
 Tao, An I-671  
 Tao, Hong I-98  
 Theera-Ampornpant, Nawanol II-82  
 Thiyagalingam, Jeyarajan II-100  
 Tian, Zhen II-140  
 Tran, Oanh Thi I-864
- Venkatesh, Svetha I-42, II-256
- Wan, Jianyi I-928  
 Wang, Bai I-348  
 Wang, Dongxu I-85, I-191  
 Wang, Fuwei I-402  
 Wang, Guang II-491  
 Wang, Hai-Qing I-335  
 Wang, Haishuai I-993  
 Wang, Hao II-482
- Wang, Haosen II-335  
 Wang, Haozheng I-1  
 Wang, Hui II-174  
 Wang, Jie II-56  
 Wang, Jun I-1, I-772, II-508  
 Wang, Kuansong I-531  
 Wang, Lei I-71  
 Wang, Mengmeng II-1  
 Wang, Mingwen I-928, II-387  
 Wang, Shiqiang II-113  
 Wang, Siyu I-604  
 Wang, Tinghua I-246  
 Wang, Wei I-29  
 Wang, Weiping I-150  
 Wang, Xiangdong I-531  
 Wang, Xiao I-163  
 Wang, Xili II-182  
 Wang, Xishun I-113  
 Wang, Xiuling II-369  
 Wang, Xuequn II-237  
 Wang, Yan II-182  
 Wang, Yonghe I-217  
 Wang, Yuchen II-343  
 Wang, Zheng I-163  
 Wang, Zhenhua I-759, II-360  
 Wang, Zhihai II-113  
 Wang, Ziwen II-56  
 Wei, Jin-Mao I-1  
 Wei, Jinmao I-772  
 Wei, Lu II-335  
 Wen, Ying I-574  
 Wobcke, Wayne II-300  
 Wong, Kok-Wai II-237  
 Wu, Fan I-361  
 Wu, Jie I-98  
 Wu, Jun I-993, I-1005  
 Wu, Xiao-Jun II-464  
 Wu, Yi-Feng II-64  
 Wu, Yiming II-491  
 Wu, Zhonghai I-915, II-455
- Xia, Chao I-310  
 Xia, Siyu I-310  
 Xiang, Ming I-177  
 Xiao, Yi I-797  
 Xie, Dongliang II-335  
 Xie, Juanying II-317  
 Xie, Nengfu I-941  
 Xie, Qiang II-149

- Xin, Yingchu II-219  
 Xiong, Deyi I-137  
 Xiran, Sun II-91  
 Xu, Jian II-491  
 Xu, Jingda II-274  
 Xu, Ming II-491  
 Xu, Ning I-671  
 Xu, Qing I-710  
 Xu, Tonglin I-1058  
 Xu, Wenqiang II-228  
 Xu, Ying I-268  
 Xu, Zenglin I-837  
 Xuan, Kangxi II-47  
 Xue, Bing I-732  
 Xue, Hui I-454  
  
 Yang, Ang II-256  
 Yang, Chengxi I-746  
 Yang, Huan II-378  
 Yang, Jufeng I-684  
 Yang, Meng I-891  
 Yang, Peng I-490  
 Yang, Tianpei I-85, II-132  
 Yang, Yang II-64  
 Yang, Yu-Bin I-126, I-335, I-477  
 Yang, Yufan I-204  
 Yang, Zhenglu I-1  
 Yang, Ziwen II-265  
 Yangming, Liu II-91  
 Yao, Dezhong I-837  
 Yao, Xingxu I-684  
 Ye, Jingjing I-851  
 Yin, He-Feng II-464  
 Yin, Kejie II-309  
 Yin, Minzhi I-545  
 Yousefnezhad, Muhammad I-1058  
 Yu Galan, Stanley II-73  
 Yu, Chao I-85, I-191  
 Yu, Gang I-772  
 Yu, Guoxian I-837, II-508  
 Yu, Shuang II-499  
  
 Zang, Liangjun I-518  
 Zeng, Qingtian I-941  
 Zeng, Qunsheng II-47  
 Zhan, Choujun II-10  
 Zhan, De-Chuan II-64  
 Zhang, Canlong II-38  
 Zhang, Changqing I-163  
 Zhang, Chaoli I-361  
 Zhang, Cheng II-274  
 Zhang, Chengwei II-132  
 Zhang, Chunxia I-878  
 Zhang, Daoqiang I-1058, I-1069  
 Zhang, De II-1  
 Zhang, Dong I-375  
 Zhang, Feifei I-204, II-326  
 Zhang, Feng II-1  
 Zhang, Hao II-228  
 Zhang, Haofeng I-503, I-588  
 Zhang, Honglei I-1005  
 Zhang, Hua-ping I-811  
 Zhang, Hui I-217  
 Zhang, Jian I-759  
 Zhang, Jianhua II-309, II-360  
 Zhang, Kai-Jun I-126  
 Zhang, Le I-310  
 Zhang, Lei I-710  
 Zhang, Li I-719  
 Zhang, Lijing II-438  
 Zhang, Ling II-19  
 Zhang, Lu I-518  
 Zhang, Mengjie I-732  
 Zhang, Minjie I-113, II-343  
 Zhang, Peng II-274  
 Zhang, Pengqing I-658, I-968  
 Zhang, Pengyu I-57  
 Zhang, Qian II-201  
 Zhang, Qiaowei I-617  
 Zhang, Quexuan I-904  
 Zhang, Xi I-811, I-878  
 Zhang, Xiang I-559  
 Zhang, Xianke I-941  
 Zhang, Xinyu I-643  
 Zhang, Yazhou I-968  
 Zhang, Yiyi II-121  
 Zhang, Yujia I-993  
 Zhang, Yujun II-201  
 Zhang, Yun I-150  
 Zhang, Yupei I-177  
 Zhang, Yuxiang I-684  
 Zhang, Zhao I-71, I-98, II-10  
 Zhang, Zhaoxiang II-210, II-247  
 Zhang, Zhuoxing I-1044  
 Zhang, Zili II-508  
 Zhang, Zixing I-574  
 Zhang, Zongzhang I-321, II-421  
 Zhao, Hailiang I-268

Zhao, Jing I-954  
Zhao, Lei I-811, I-878  
Zhao, Mingbo II-10  
Zhao, Peng I-1018, II-473  
Zhao, Weixuan I-759  
Zhao, Zhixuan II-473  
Zhao, Zhonghua II-369  
Zheng, Fa I-454  
Zheng, Hanfeng II-438  
Zheng, Ning II-491  
Zheng, Wu II-210  
Zheng, Yan I-797, II-421  
Zhenlei, Wang II-91

Zhou, Liang II-149  
Zhou, Lingli I-503  
Zhou, Tao II-38  
Zhou, Xiaosong II-430  
Zhou, Xiaoyu II-430  
Zhou, Xinyu I-928, II-387  
Zhou, Yuan I-310  
Zhu, Pengfei I-163  
Zhu, Wenxuan I-85  
Zhu, Xianyi I-797  
Zhuang, Bojin II-56  
Zhuang, Yuan I-980  
Zhuge, Wenzhang I-98